Краткое содержание

Оглавление

Введение

Книга, которую вы держите в руках, содержит сведения о таких малоизвестных возможностях Windows, как работа с программой `rundll32.exe`, с реестром операционной системы, а также описание оснасток и работы с консолью управления `mmc.exe`. Кроме того, с ее помощью вы сможете лучше узнать предназначение отдельных папок, файлов журнала и библиотек, используемых компонентами операционной системы Windows XP. В книге будут также рассмотрены некоторые примеры создания сценариев сервера сценариев Windows и INF-файлов.

Эта книга предназначена скорее для опытных пользователей и администраторов, не использующих Active Directory, которым интересно узнать о нестандартных возможностях Windows, а также о некоторых простых способах ее взлома. Однако она может быть интересна и простым пользователям. В общем, автор охарактеризовал бы круг читателей так: для всех, кто хочет знать больше.

Теперь, когда общие вопросы рассмотрены, скажем еще несколько слов о темах, которые будут описаны в книге.

- **Часть 1. «Программа rundll32.exe»** — первая часть книги посвящена гениальной программе от Microsoft `rundll32.exe`, входящей в стандартную поставку Windows. Функциональность этой программы можно охарактеризовать так — она предназначена для выполнения библиотечных и API-функций программ напрямую из Windows, без участия самой программы. К сожалению, об этой программе уже стали забывать даже сами программисты Microsoft, иначе как объяснить тот факт, что с ее помощью можно обойти некоторые ограничения, которые устанавливаются с помощью административных шаблонов Windows. Но об этом далее.

  Конечно, это описание ничего пока не говорит, но автор уверен, что после прочтения первой части книги вы разделите его восхищение этой программой.

- **Часть 2. «Реестр Windows XP»** — вторая часть посвящена системному реестру Windows XP. В ней будут рассмотрены возможности настройки операционной системы, доступ к которым нельзя получить с помощью диалоговых окон операционной системы. Описаны также некоторые настройки конфигурации системы, с помощью которых можно повысить ее общую производительность.

  Вторая часть предназначена пользователям, которые не понаслышке знают о реестре. Если же вы ничего о нем не знаете, то вам будет довольно трудно понять способ работы с ним на основе только этой части, но возможно.

- **Часть 3. «Консоль управления Microsoft»** — третья часть описывает консоль управления Microsoft. Это программа, которая содержит всю функциональную составляющую Windows, направленную на администрирование системы. Здесь рассмотрены следующие вопросы: общая работа с консолью управления, работа с отдельными ее оснастками, параметры реестра, которые изменяются оснастками и некоторыми административными шаблонами, а также более узконаправленные вопросы работы отдельных оснасток консоли.
- **Часть 4. «Другие возможности Windows XP»** — в четвертой части рассмотрены другие интересные возможности операционной системы, о которых вы можете не знать. Здесь описаны принципиальные различия между версиями операционной системы Windows XP, некоторые интересные особенности INF-файлов, возможности нестандартной настройки Windows XP, сказано несколько слов о безопасности компьютера и роли в этой безопасности учетной записи системы. В этой части рассмотрены также каталоги файловой системы Windows XP, и вы сможете узнать, что же в них хранится и для чего они вообще нужны.

Кроме того, книга содержит шесть приложений, из которых вы узнаете о наиболее интересных ActiveX-объектах, используемых в операционной системе, а также о содержимом некоторых справочных файлов, входящих в состав операционной системы.

Вместе с книгой поставляется база данных, содержащая описанные в издании параметры реестра.

От издательства

Ваши замечания, предложения, вопросы отправляйте по адресу электронной почты gurski@minsk.piter.com (издательство «Питер», компьютерная редакция).

На сайте издательства http://www.piter.com вы найдете подробную информацию о наших книгах.

Часть 1

Программа rundll32.exe

Глава 1

Работа с оболочкой

- CPL-файлы
- ActiveX-объекты
- Драйверы
- Библиотеки

`Rundll32.exe` — это небольшая программа, поставляемая со всеми версиями Windows.

Она была создана программистами Microsoft для своих нужд и способна выполнять любую Wind32 API-функцию, описанную в библиотеке, как будто эта функция вызывается из программы или сама является полноценной программой. `Rundll32.exe` создавалась для вызова различных функций из файлов сценариев или INF-файлов, с ее помощью можно выполнить очень много интересных и полезных трюков, описание которых приведено в данной главе.

Перед описанием команд, доступных при работе с `rundll32.exe`, хотелось бы еще сузить рамки функций, которые необходимо понимать под «любыми» функциями, выполняемыми с помощью данной программы.

Несмотря на то, что программа `rundll32.exe` способна выполнять любые функции, не все функции смогут быть выполнены — все дело в том, что некоторые из функций могут ожидать каких-то специальных параметров, не являющихся статическими и получаемых с помощью других функций.

Еще хуже ситуация может обстоять в том случае, когда функция не принимает никаких параметров, но при своей работе использует содержимое регистров или памяти компьютера. В этом случае вызов данной функции может быть подобен игре в кости — если вам повезет и все необходимые данной функции параметры окажутся корректными, то она сможет выполниться.

Синтаксис программы `rundll32.exe` довольно прост:

rundll32 «[illegible]», «[illegible]», «[illegible]»

Но как же можно вызвать данную программу? Помимо использования вызова в коде сценария, о чем будет рассказано в следующих главах книги, программу можно вызвать и с помощью диалога Запуск программы (Пуск ▸ Выполнить) или в поле команды при создании ярлыка. Например, в окне Запуск программы введите команду `rundll32 shell32.dll, ShellAboutA` и вы сможете увидеть диалог, подобный приведенному на рис. 1.1.

ВНИМАНИЕ

Если регистр, в котором вы вводите команду, для названия параметров функции и библиотек не важен, то для названия функции следует внимательно следить за его написанием как с точки зрения ошибок, так и с точки зрения регистра каждого отдельного символа, иначе программа rundll32.exe не сможет найти необходимую вам функцию. Например, если вы вместо названия функции ShellAboutA введете название ShellaboutA, то rundll32.exe выведет сообщение о невозможности вызова функции, так как ее нет в соответствующей библиотеке.

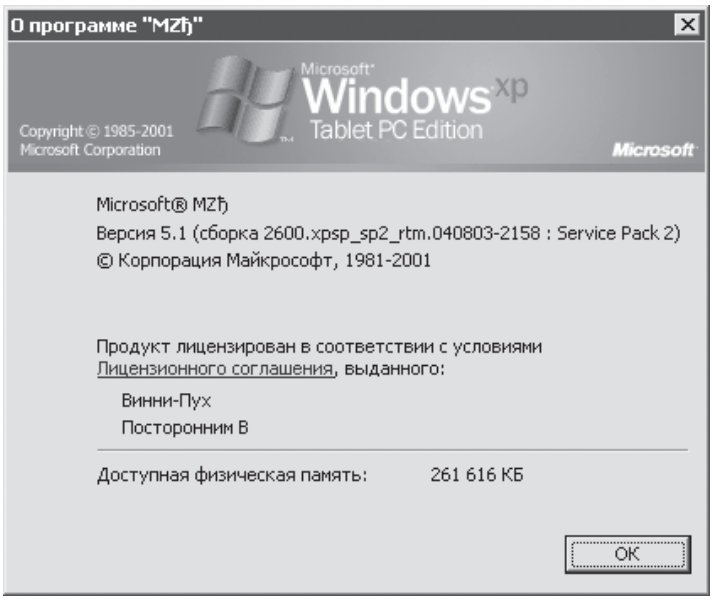


Рис. 1.1. Окно с информационными сведениями об операционной системе

CPL-файлы

Из нескольких предыдущих абзацев вы узнали практически все теоретические выкладки, которые необходимы для работы с программой `rundll32.exe`. Сейчас же будут перечислены те возможности, которые данная программа может предоставить пользователю. Начнем с описания функций CPL-файлов.

ПРИМЕЧАНИЕ

Перед тем как будут рассмотрены различные функции, которые могут выполняться с помощью программы rundll32.exe, хотелось бы уточнить способ записи некоторых функций. Чуть выше рассмотрена команда, выполняющая функцию ShellAboutA. Как можно заметить, эта функция заканчивается каким-то совершенно ненужным символом — A. Он говорит системе о том, что функция использует для своей работы символы кодировки ASCII (использование одного байта для представления одного символа). Существует еще один символ, которым может заканчиваться функция — символ W. Он говорит системе о том, что функция использует для своей работы символы в кодировке Unicode (для представления одного символа используются два байта). Так вот, эти символы совершенно не обязательны (хотя в очень редких случаях они необходимы) — если вы введете название функции без последнего символа (A или W), то система сама попробует определить, какая кодировка должна использоваться.

Хотя раньше говорилось, что программа `rundll32.exe` может вызывать функции библиотек, на самом деле ее возможности намного обширнее — она может

вызвать функцию отовсюду, где она указана, будь то библиотека, CPL-файлы или сам исполняемый файл программы.

CPL-файл — это специальный файл (их еще называют апплетами), являющийся диалоговым окном, которое вызывается с помощью Панели управления и предназначено для настройки какого-то отдельного компонента операционной системы. Большую часть ссылок на CPL-файлы можно встретить в папке Панель управления, которую можно открыть либо с помощью меню Пуск, либо с помощью вызова команды `control` или команды `shell:ControlPanelFolder`. Но операционная система Windows позволяет вызывать CPL-файлы и с помощью ввода их названия в окне Запуск программы (в этом случае расширение CPL указывать обязательно).

В контексте данной книги содержимое CPL-файлов рассматриваться не будет, предполагается, что читатель уже пользовался ими. Исключением могут быть только те случаи, когда соответствующая команда `rundll32.exe` зависит от установки тех или иных параметров, доступ к которым можно получить из CPL-файла.

Access.cpl

Апплет предназначен для редактирования различных специальных параметров настройки клавиатуры, звука, мыши или оболочки Windows. Все эти параметры разрабатывались программистами Microsoft специально для людей с ограниченными физическими возможностями, хотя некоторые из представленных в апплете параметров могут быть полезны и в повседневной работе всех пользователей операционной системы Windows XP.

Чтобы вызвать данный апплет с помощью программы `rundll32.exe`, необходимо использовать команду `rundll32 Access.cpl, DebugMain`. Вот, в принципе, и все команды, которые поддерживает апплет `access.cpl`.

Appwiz.cpl

Эта команда открывает диалоговое окно Установка и удаление программ. С его помощью можно как удалять или изменять составляющую различных установленных в системе программ, так и заменять составляющие компоненты самой операционной системы.

В отличие от рассмотренного ранее апплета `access.cpl`, апплет `appwiz.cpl` не позволяет вызвать себя с помощью команды программы `rundll32.exe`. Зато он дает возможность выполнить некоторые действия, другими способами не выполняющиеся. Но перед тем как рассмотреть возможные действия, которые разрешает выполнить апплет `appwiz.cpl`, скажем несколько слов о работе самого апплета — это поможет нам в понимании работы рассматриваемых далее команд.

Как уже сказано, апплет `appwiz.cpl` содержит список всех установленных в системе приложений — для его отображения применяется список Установленные

программы (рис. 1.2). Но как формируется данный список? Можно подумать, что это происходит при каждом запуске апплета `appwiz.cpl` путем сканирования файловой системы Windows, но все намного проще. На самом деле данный список расположен в реестре Windows. Для его хранения используется ветвь `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall`, включающая набор разделов, каждый из которых определяет один элемент списка Установленные программы. Названия этих разделов идентифицируют установленную программу для апплета `appwiz.cpl`, но при этом не используются в списке Установленные программы. Для указания названия элемента в списке применяется строковый параметр `DisplayName`, расположенный в каждом из разделов рассматриваемой ветви (см. рис. 1.2).

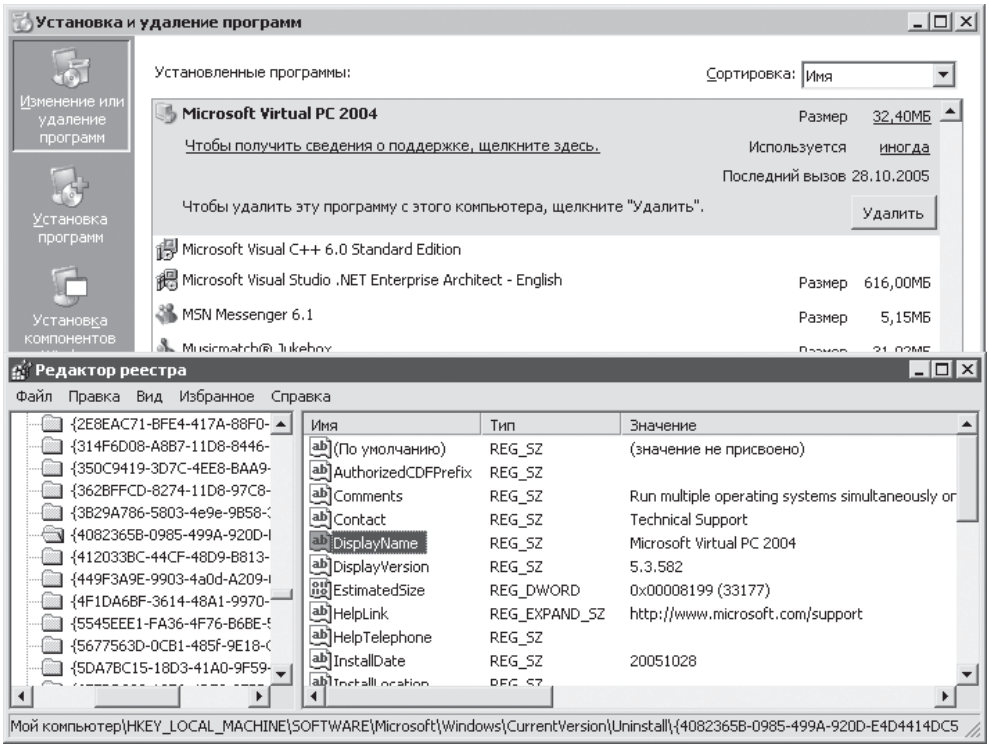


Рис. 1.2. Формирование списка Установленные программы апплета appwiz.cpl

Здесь не будет полностью рассказано о параметрах разделов ветви реестра Windows `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall` — это не является главной темой книги. Поэтому читателям, которым интересна данная тема, рекомендуется купить дополнительную книгу, посвященную только ей. Мы же сейчас займемся тем, для чего и был затеян рассказ о способе хранения списка установленных программ апплетом `appwiz.cpl`, — описанием команды `rundll32.exe`, с помощью которой можно удалить программу, указанную в данном списке.

Для того чтобы удалить программу, ссылка на которую содержится в ветви реестра HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall, необходимо воспользоваться следующей командой:

rundll32 appwiz.cpl, WOW64Uninstall_RunDLL „„«□□□□□□□□ □□□□□□□□ □ □□□□□□ □□□□□□□□
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall,
□□□□□□□□□ □□□□□□□□□□□□□□□□□□□□□□□□»

Например, если вы хотите удалить программу EVEREST Professional v1.50, раздел которой в рассматриваемой ветви назван EVEREST Professional_is1 (рис. 1.3), то необходимо использовать следующую команду:

rundll32 appwiz.cpl, WOW64Uninstall_RunDLL ,,,EVEREST Professional_is1

После ее вызова запустится команда удаления, которая записана в параметре строкового типа UninstallString раздела ветви реестра EVEREST Professional_is1 (рис. 1.3).

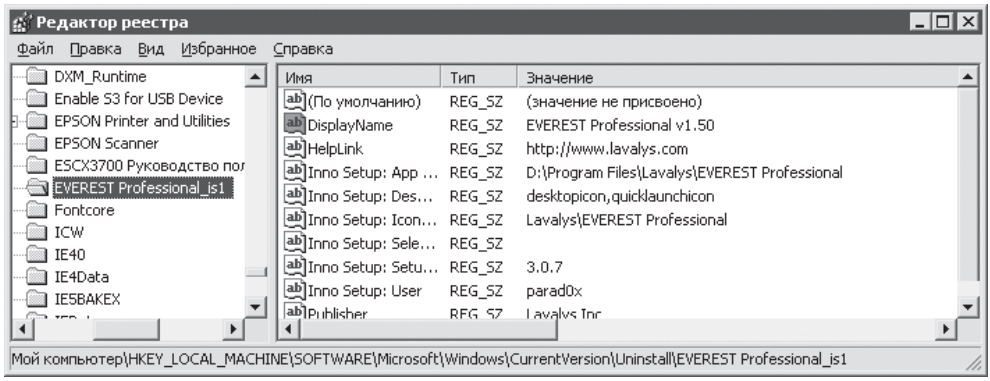


Рис. 1.3. Раздел удаляемой программы назван EVEREST professional_is1

Но, кроме удаления программ, апплет appwiz.cpl позволяет выполнять некоторые другие интересные команды, используемые оболочкой Windows. Первой из них является команда rundll32 appwiz.cpl, ConfigStartMenu, позволяющая очистить содержимое вашего меню Пуск. После ее вызова перед вами появится диалог, отображенный на рис. 1.4. С помощью этого диалога можно удалить те из элементов меню Пуск, которые находятся в папке %userprofile%\Ãëàâíîå ìåíþ (такие папки, как Администрирование, Стандартные или Автозагрузка удалить нельзя).

Другой командой, которую можно вызвать благодаря апплету appwiz.cpl, является команда rundll32 appwiz.cpl, NewLinkHere «ïóòü ê êàòàëîãó». С ее помощью можно создать ярлык в каталоге, который вы укажете. Для этого будет вызван специальный мастер создания ярлыков. Доступ к нему можно также получить из списка Создать контекстного меню Рабочего стола или папки, если выбрать в этом списке команду Ярлык.

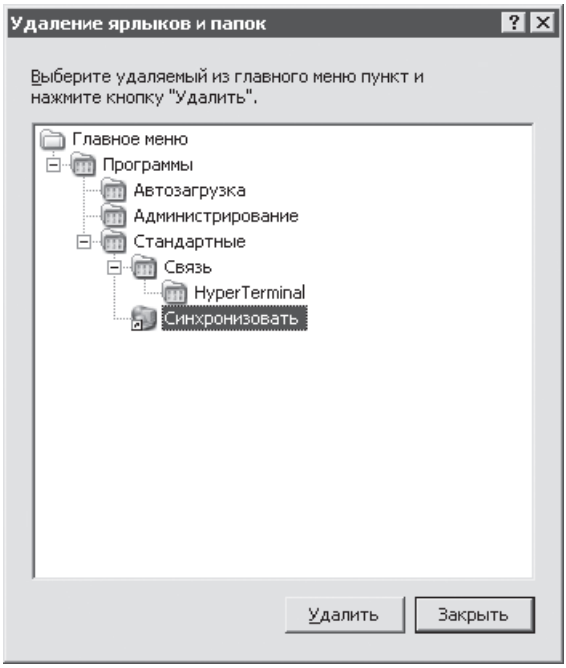


Рис. 1.4. Результат выполнения команды rundll32 appwiz.cpl, ConfigStartMenu

Bthprops.cpl

`Bthprops.cpl` — это апплет, предназначенный для настройки работы и взаимодействия устройств, подключаемых при помощи беспроводного соединения Bluetooth. По умолчанию, если в системе не установлено (или не зарегистрировано) ни одного устройства, работающего с помощью Bluetooth, то апплет `bthprops.cpl` и все настраиваемые им параметры недоступны. Если же вы хотите посмотреть, что это за апплет, то можно попытаться воспользоваться некоторыми командами `rundll32.exe`, использующими файл апплета для своей работы. К сожалению, с помощью таких команд нельзя вызвать само окно апплета, но зато можно воспользоваться некоторыми функциями, которые оно предоставляет. Например, можно вызвать агента подключений Bluetooth, с помощью которого можно подключить устройство Bluetooth, передать или отправить файлы, а также настроить конфигурацию уже подключенных устройств Bluetooth. Для этого необходимо выполнить несколько действий. Во-первых, нужно присвоить `DWORD`-параметру `Notification Area Icon`, расположенному в ветви системного реестра `HKEY_CURRENT_USER\Control Panel\Bluetooth`, значение, равное `1`. Это необходимо для того, чтобы мы могли взаимодействовать с агентом подключений посредством его значка, отображаемого на Панели задач. Во-вторых, необходимо воспользоваться командой `rundll32.exe` для запуска агента: `rundll32 bthprops.cpl ,,,BluetoothAuthenticationAgent`. После ввода данной команды на Панели задач в области уведомлений появится значок, подобный изображенному на рис. 1.5.

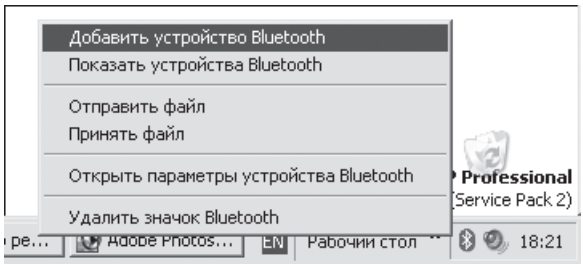


Рис. 1.5. Результат выполнения команды rundll32 bthprops.cpl ,,,BluetoothAuthenticationAgent

ПРИМЕЧАНИЕ

После закрытия агента подключений значение параметра Notification Area Icon автоматически становится равным 0.

Еще одной возможностью, которую предоставляет апплет `bthprops.cpl`, является возможность вызова диалогового окна свойств подключенного устройства Bluetooth. Для этого используется следующая команда: `rundll32 bthprops.cpl ,,,BluetoothDisplayDeviceProperties`. Эта команда будет вызывать диалог, подобный приведенному на рис. 1.6.

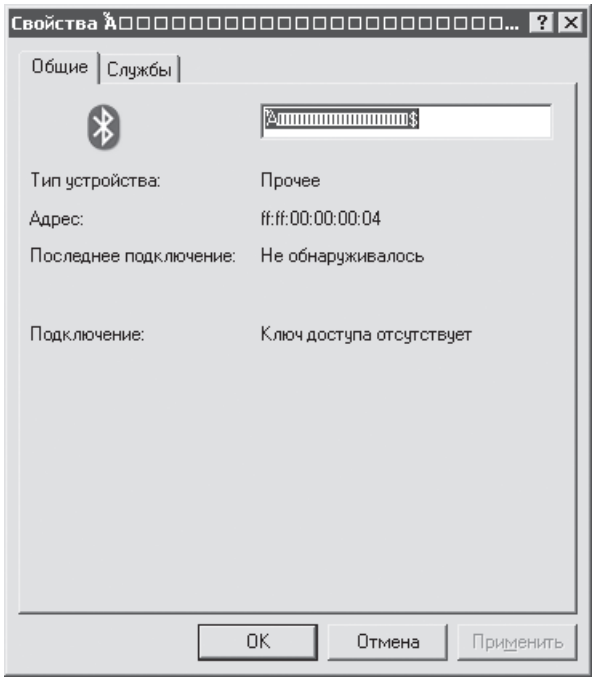


Рис. 1.6. Результат выполнения команды rundll32 bthprops.cpl ,,,BluetoothDisplayDeviceProperties

ПРИМЕЧАНИЕ

Аналогичные команды можно применить и для апплета irprops.cpl. В этом случае нужно в команде rundll32.exe вместо апплета bthprops.cpl указать апплет irprops.cpl. Несмотря на то, что используются два разных апплета, результат команд будет один и тот же.

Desk.cpl

Данный апплет знаком, наверное, каждому пользователю Windows — именно он и является диалогом Свойства: Экран, доступ к которому можно получить, если выбрать команду Свойства из контекстного меню Рабочего стола. С помощью данного апплета можно выполнить такие действия, как настройка некоторых параметров оболочки Windows, изменение обоев Рабочего стола, изменение заставки и параметров энергопотребления, изменение разрешения, глубины цвета или частоты обновления экрана.

Конечно, всех этих действий командами `rundll32.exe` заменить не получится, но некоторые из них можно выполнить и с помощью `rundll32`. Рассмотрим наиболее интересные команды.

- `rundll32 desk.cpl, DisplayTestSettingsW` — позволяет протестировать настройки монитора, во время чего монитор сначала отключается, а потом включается с новыми настройками. Конечно, новые настройки придется изменять посредством реестра, а вообще, эта команда может быть использована не только для тестирования настроек, но и для скрытия деструктивных или других функций сценария.
- `rundll32 desk.cpl, InstallScreenSaver «ïóòü ê íîâîìó ôàéëó çàñòàâêè»` — с помощью данной команды можно автоматически сменить используемую по умолчанию заставку, отображаемую при простое системы в течение некоторого времени. Эту команду `rundll32.exe` можно также использовать и без пути к файлу заставки. В этом случае она будет отображать окно Свойства: Экран, открытое на вкладке Заставка.
- `rundll32 desk.cpl, UpdateUIfontsDueToDPIchange` — это очень опасная команда, особенно если применять ее без параметров. Ее действия непредсказуемы, но в основном они направлены на обновление параметров настройки оболочки из ветвей реестра `HKEY_CURRENT_USER\Control Panel\Desktop` и `HKEY_CURRENT_USER\Control Panel\Colors`. При этом если не указывать параметры вызова команды, то значения параметров, которые она устанавливает в приведенных ветвях реестра, являются случайными.

  Конечно, приведенное выше описание команды условно и наигранно, по этой причине для примера результата действий команды хотелось бы представить рис. 1.7. Приведенные на рисунке настройки оболочки были получены после нескольких последовательных вызовов команды `rundll32 desk.cpl, UpdateUIfontsDueToDPIchange`.

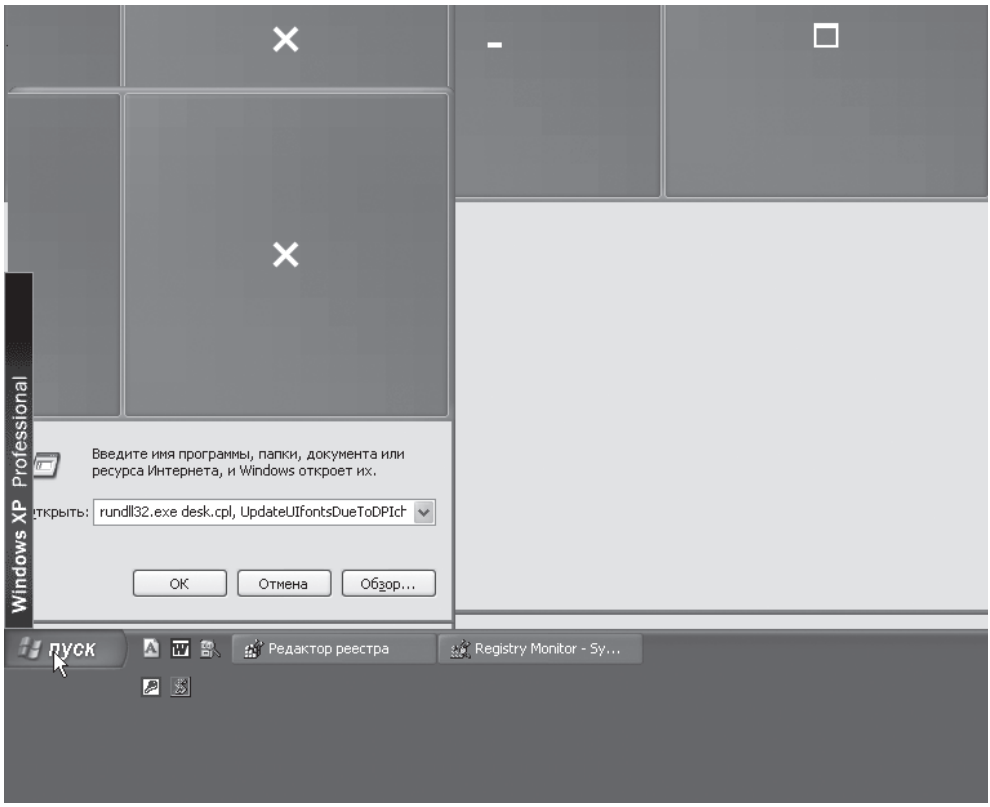


Рис. 1.7. Оказывается, поиск команд rundll32 — это очень опасное занятие

Firewall.cpl

Апплет служит для настройки встроенного брандмауэра Windows (работает ли брандмауэр, за работой в сети каких программ он не следит). Окно этого апплета можно вызвать с помощью команды `rundll32 firewall.cpl, ShowControlPanel`.

Кроме вызова окна настройки брандмауэра Windows, файл `firewall.cpl` позволяет вызвать другое окно — окно извещения о том, что функциональность заданной программы будет ограничена. Для этого применяется следующая команда: `rundll32 firewall.cpl, ShowNotificationDialog «íàçâàíèå ïðîãðàììû»`. Например, результат вызова команды `rundll32 firewall.cpl, ShowNotificationDialog "c:\windows\system32\cmd.exe"` можно увидеть на рис. 1.8 (можно просто указать название программы — `cmd.exe`, в этом случае в строке `Имя` вместо описания программы будет отображено ее название).

Вызванное окно является мнимым, то есть его вызов ни к какому результату не приведет и никак не повлияет на возможность работы указанной в диалоговом окне программы.

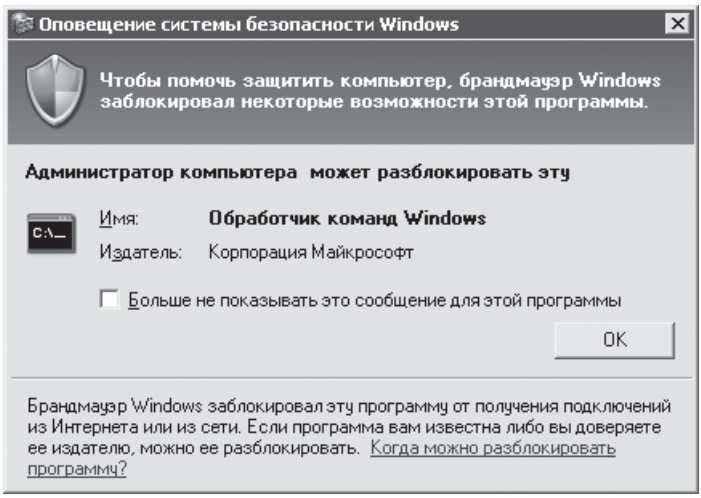


Рис. 1.8. Результат выполнения команды rundll32 firewall.cpl, ShowNotificationDialog "c:\windows\system32\cmd.exe"

Inetcpl.cpl

Апплет является диалогом Свойства: Интернет, предназначенным для настройки параметров конфигурации браузера Internet Explorer. Доступ к этому диалогу можно получить как через Панель управления, так и с помощью команды Свойства обозревателя меню Сервис браузера Internet Explorer. Уже стало традицией, что доступ к апплету можно получить не только при помощи его названия, но и используя команду `rundll32.exe` — в данном случае для этого применяется команда `rundll32 inetcpl.cpl, LaunchInternetControlPanel`. Кроме этой команды, апплет `inetcpl.cpl` поддерживает довольно много команд `rundll32.exe`, поэтому для их описания воспользуемся списком.

- `rundll32 inetcpl.cpl, DisplayPopupWindowManagementDialog` — позволяет вывести на экран диалог Параметры блокировки всплывающих окон, предназначенный для настройки сайтов, на которые не будут действовать параметры блокировки. При этом в поле Адрес разрешенного Веб-узла будет установлено нечитаемое значение. К сожалению, изменить это значение невозможно. С помощью данного окна можно также определить, будут ли подаваться звуковые сигналы при блокировке всплывающего окна и будет ли отображаться в строке статуса браузера панель информации при блокировке окна.
- `rundll32 inetcpl.cpl, DllInstall` — из следующей части, описывающей интересные ветви и параметры реестра, вы узнаете о том, что содержимое списка на вкладке Дополнительно окна Свойства: Интернет хранится в реестре. При этом его можно редактировать или вообще удалить, чтобы пользователь не смог изменить настройки браузера. Если вы все-таки удалили содержимое списка на вкладке Дополнительно, то, выполнив данную команду `rundll32.exe`, вы всегда сможете его восстановить (восстанавливается список по умолчанию, то есть если вы добавили к списку свои элементы, то они будут утеряны).

- `rundll32 inetcpl.cpl, LaunchConnectionDialog` — позволяет вызвать окно Свойства: Интернет, открытый на вкладке Подключения.
- `rundll32 inetcpl.cpl, LaunchPrivacyDialog` — если с помощью предыдущей команды можно было открыть диалог Свойства: Интернет на вкладке Подключения, то с помощью этой команды можно открыть его на вкладке Конфиденциальность.
- `rundll32 inetcpl.cpl, LaunchSecurityDialogEx` — позволяет открывать вкладку Безопасность как независимый от диалога Свойства: Интернет элемент системы (рис. 1.9). Другими словами, если две предыдущие команды не смогли бы выполниться в случае, когда с помощью административных шаблонов запрещено отображать соответствующие вкладки или сам диалог, то эта команда не зависит от настроек групповых политик.

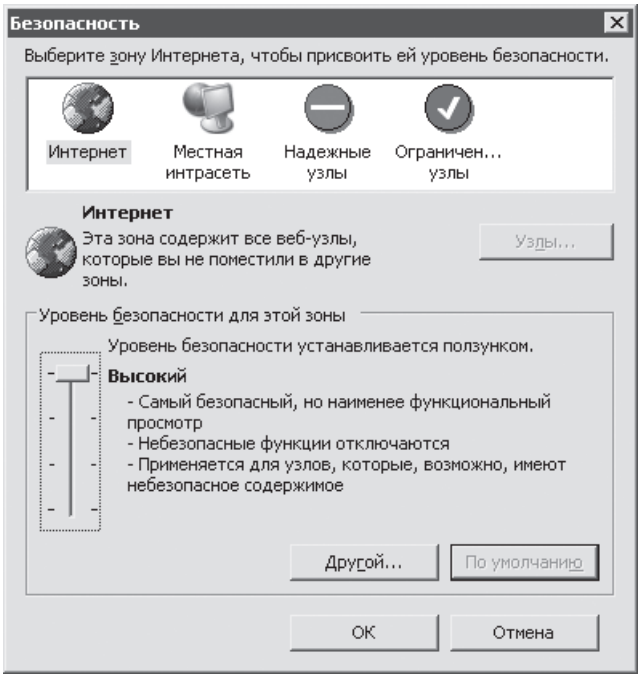


Рис. 1.9. Результат выполнения команды rundll32 inetcpl.cpl, LaunchSecurityDialogEx

- `rundll32 inetcpl.cpl, LaunchSiteCertDialog` — благодаря этой команде можно вывести диалог Сертификаты, с помощью которого вы сможете импортировать, экспортировать или просто просмотреть список личных сертификатов, а также доверенных издателей, доверенных центров сертификатов и многое другое.
- `rundll32 inetcpl.cpl, OpenLanguageDialog` — если два предыдущих диалога не влияли ни на одну программу операционной системы, то диалог, открываемый с помощью этой команды, отображается при нажатии кнопки Языки на вкладке Общие окна Свойства обозревателя.

- `rundll32 inetcpl.cpl, SiteCert_RunFromCmdLine «ïóòü ê ôàéëó êîðíåâîãî ñåðòèôèêàòà»` — применяется для запуска установки файла корневого сертификата.

Joy.cpl

С помощью данного апплета можно отобразить маленькое окно настройки или установки нового игрового манипулятора — джойстика, руля и т. д. Поскольку диалоговое окно данного апплета действительно маленькое, неудивительно, что оно поддерживает только одну команду `rundll32.exe` — `rundll32 joy.cpl, ShowJoyCPL`.

Mmsys.cpl

В этом разделе будут рассмотрены несколько команд `rundll32.exe`, предназначенных для отображения той или иной части диалога Свойства: Звук и аудиоустройства. Этот диалог используется для настройки событий системы, которые требуют звукового сопровождения, а также для настройки некоторых возможностей колонок и микрофона. Чтобы отобразить диалог Свойства: Звук и аудиоустройства с помощью `rundll32.exe`, необходимо воспользоваться командой `rundll32 mmsys.cpl, ShowFullControlPanel`. Можно также использовать команду `rundll32 mmsys.cpl, ShowDriverSettingsAfterFork`. В этом случае отображаемый диалог будет открыт на вкладе Оборудование.

Существует возможность вызова отдельных диалогов, используемых для построения диалога Свойства: Звук и аудиоустройства. Например, можно отобразить диалог Свойства аудио (рис. 1.10), являющийся вкладкой Аудио диалога Свойства: Звук и аудиоустройства. Для этого применяется команда `rundll32 mmsys.cpl, ShowAudioPropertySheet`.

Nusrmgr.cpl

Апплет отображает новый диалог Windows XP, предназначенный для настройки учетных записей пользователей. Этот диалог имеет очень много интересных возможностей, которые вы можете использовать в том случае, если аутентификация пользователя при его входе в систему на вашем компьютере выполняется с помощью нового стиля оформления Windows XP. Например, вы можете изменить изображение, которое будет выводиться напротив учетной записи конкретного пользователя (рис. 1.11). С помощью этого диалога можно настроить параметры паспорта .NET, а также имя, пароль или тип учетной записи конкретного пользователя.

К сожалению, с помощью команд `rundll32.exe` нельзя вызвать диалог Учетные записи пользователей или одну из его составляющих, хотя вы можете воспользоваться параметрами вызова апплета `nusrmgr.cpl`, описанными в приложении 2, чтобы вызвать ту или иную страницу данного диалога.

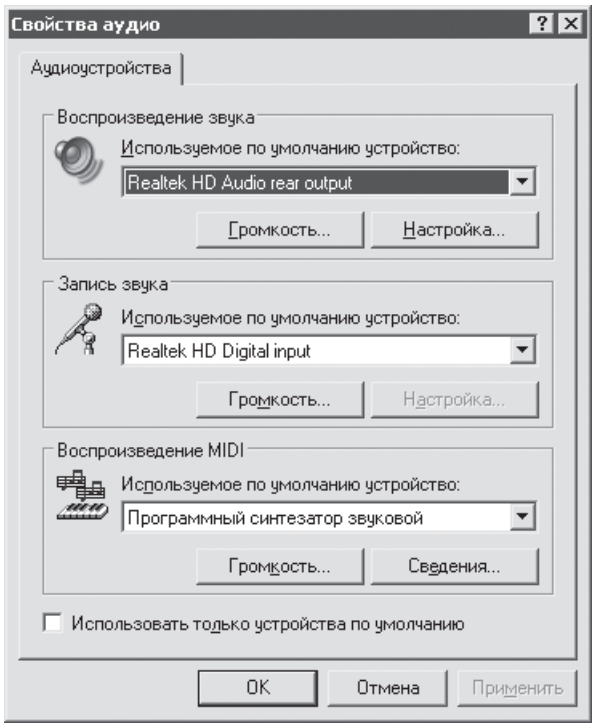


Рис. 1.10. Результат выполнения команды rundll32 mmsys.cpl, ShowAudioPropertySheet

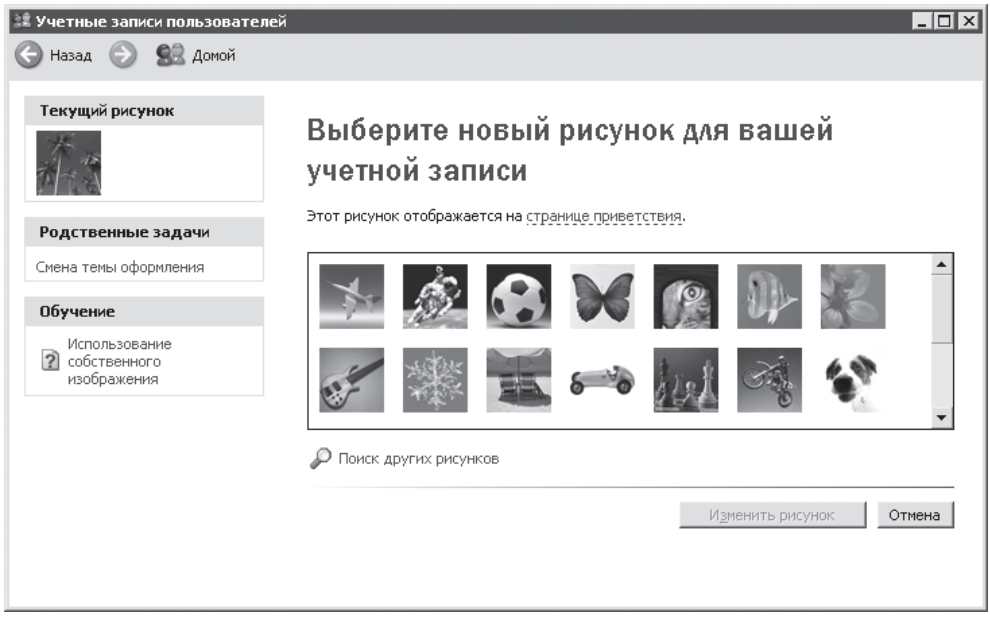


Рис. 1.11. Изменение изображения, выводимого напротив учетной записи пользователя

Несмотря на то, что данное окно нельзя вызвать с помощью команд `rundll32.exe`, некоторые возможности апплет `nusrmgr.cpl` все-таки предоставляет. Например, с его помощью можно удалить из реестра все сведения о диалоге Учетные записи пользователей, в результате чего этот диалог больше не будет работать. Для этого применяется команда `rundll32 nusrmgr.cpl, DllUnregisterServer`. Конечно, произведенные изменения обратимы. Например, чтобы восстановить в реестре все сведения о диалоге Учетные записи пользователей, нужно воспользоваться командой `rundll32 nusrmgr.cpl, DllRegisterServer`. Теперь диалог Учетные записи пользователей опять будет работать.

Wuaucpl.cpl

Это последний апплет команды `rundll32.exe`, который будет рассмотрен. Он предназначен для настройки автоматического обновления Windows и определяет способы его работы: будет ли оно работать; если все-таки будет работать, то в какое время будет происходить подключение к сайту обновления. Апплет также определяет параметры загрузки и установки обновлений. Чтобы отобразить диалоговое окно апплета `wuaucpl.cpl` с помощью `rundll32.exe`, необходимо воспользоваться командой `rundll32 wuaucpl.cpl, ShowAUControlPanel`.

Кроме отображения окна апплета, существует еще одна возможность — отображение справки по использованию автоматического обновления. Для этого применяется команда `rundll32 wuaucpl.cpl, OpenAUHelpTopic`. При этом следует учитывать, что данная команда работает корректно, только если диалог автоматического обновления отображается, иначе, возможно, придется вызвать ее несколько раз.

Существует возможность установки переключателя Загружать обновления; пользователь назначит время установки в диалоговом окне автоматического обновления Windows. Для этого применяется следующая команда: `rundll32 wuaucpl.cpl, SaveAUApprovalOptions`. Но она не всегда работает. Скажем даже, что в большинстве случаев она вызывает ошибку, но если несколько раз подряд вызвать данную команду, не закрывая сообщения об ошибках предыдущих команд, то существует большая вероятность того, что переключатель будет установлен. Вы спросите, зачем нужно столько мучиться? Все дело в том, что команда выполняется даже тогда, когда соответствующий диалог (или его настройки) заблокирован администратором (рис. 1.12) (конечно, если у вас есть права на изменение параметров ветви реестра `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\WindowsUpdate\Auto Update`).

Команда, которой также можно воспользоваться благодаря апплету `wuaucpl.cpl`, — `rundll32 wuaucpl.cpl, SaveConfigVerToRegistry`. Она устанавливает значение DWORD-параметра `ConfigVer`, расположенного в ветви системного реестра `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\WindowsUpdate\Auto Update`, равным 1. К счастью, эта команда работает чаще, чем ей предшествующая.

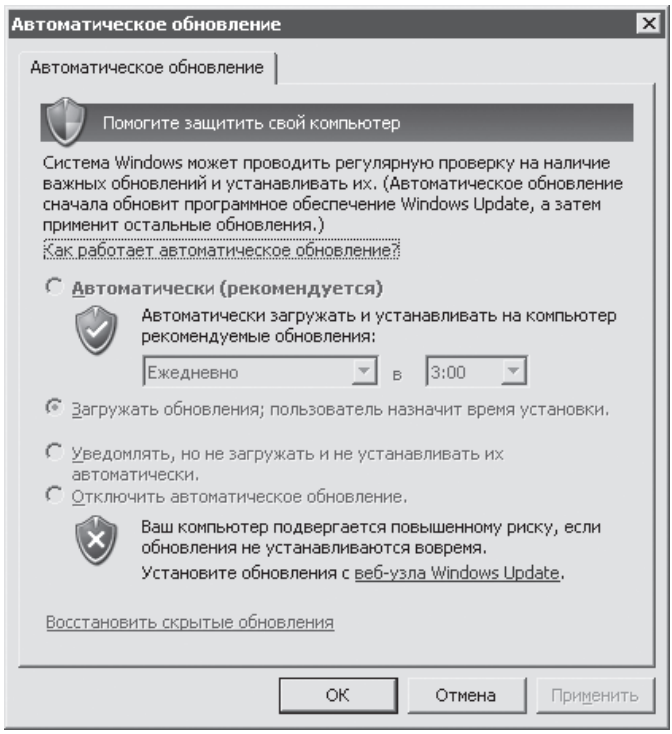


Рис. 1.12. Установка второго переключателя, несмотря на блокировку апплета

ActiveX-объекты

ActiveX-объекты — это специальные программы, которые не могут быть выполнены непосредственно двойным щелчком кнопкой мыши на файле ActiveX-объекта, но могут быть выполнены под руководством операционной системы Windows. Для этого применяются либо операции вставки и внедрения, либо команда `rundll32.exe`. Да, несмотря на то, что ActiveX-объекты не являются полноценными программами, некоторые из них также возможно вызвать как программы. Например, попробуем выполнить команду `rundll32 amovie.ocx, RunDll /play /close E:\music\Ìóëüòôèëüìû\Ìóçûêà çâåçä\5.wma` (не забудьте изменить путь и имя музыкального файла на свои). Результат выполнения этой команды можно увидеть на рис. 1.13.

Рис. 1.13. Пример вызова ActiveX-объекта

ПРИМЕЧАНИЕ

Плюсом таких вызовов является то, что каждый вызванный ActiveX-объект выполняется как отдельная программа. Другими словами, если вы несколько раз вызовете один и тот же ActiveX-объект, то будет открыто несколько ActiveX-объектов (в данном случае проигрывателей), а не только один.

Рассмотрим приведенный выше вызов ActiveX-объекта детальней: `rundll32 amovie.ocx, RunDll /play /close E:\music\Ìóëüòôèëüìû\Ìóçûêà çâåçä\5.wma`. Как можно заметить, он практически ничем не отличается от вызова функций из библиотек — `amovie.ocx` является файлом, из которого берется функция, `RunDll` является самой функцией, а путь к музыкальному файлу — ее аргументом. Кроме пути к музыкальному файлу, используется еще два аргумента функции — `/play` и `/close`. Первый из них говорит о том, что данный ActiveX-объект необходимо вызвать и начать проигрывать, а второй аргумент означает, что после выполнения своей задачи ActiveX-объект должен быть автоматически закрыт.

Кроме функции `RunDll`, ActiveX-объект `amovie.ocx` (да и большинство других ActiveX-объектов) поддерживает еще две функции — `DllRegisterServer` и `DllUnregisterServer`. Первая из них регистрирует ActiveX-объект в реестре операционной системы, а вторая удаляет из реестра всю информацию об ActiveX-объекте.

ПРИМЕЧАНИЕ

Даже если ActiveX-объект не будет зарегистрирован в системе, его все равно возможно вызвать с помощью функции RunDll или другой подобной функции.

ActiveX-объект `amovie.ocx` также поддерживает функцию `LoadFilterGraph`, которая используется для загрузки графического фильтра, но, по мнению автора, данной функции нельзя передать сам путь к фильтру.

Теперь вкратце рассмотрим функции некоторых ActiveX-объектов, которые могут находиться в вашей операционной системе (не забывайте, что эти ActiveX-объекты, как и все остальные, поддерживают и такие функции, как `DllRegisterServer` и `DllUnregisterServer`).

- `rundll32 HHCTRL.OCX, doWinMain` — выполнение этой команды приводит к вызову диалога HTML Help, приведенного на рис. 1.14. Содержимое диалога (файлы `connect.inf` и т. д.) находится в каталоге `%userprofile%\MZ•`. Если же данная папка отсутствует в системе, то будет выводиться ошибка о невозможности загрузки страницы.

  Этот ActiveX-объект расположен в каталоге `%systemroot%\SYSTEM32`.

- `rundll32 msdxm.ocx, RunDll /play /stop «ïóòü ê ìóçûêàëüíîìó ôàéëó»` — вызов данной функции воспроизводит указанный музыкальный файл с помощью Проигрывателя Windows Media.

  Этот ActiveX-объект расположен в каталоге `%systemroot%\SYSTEM32`.

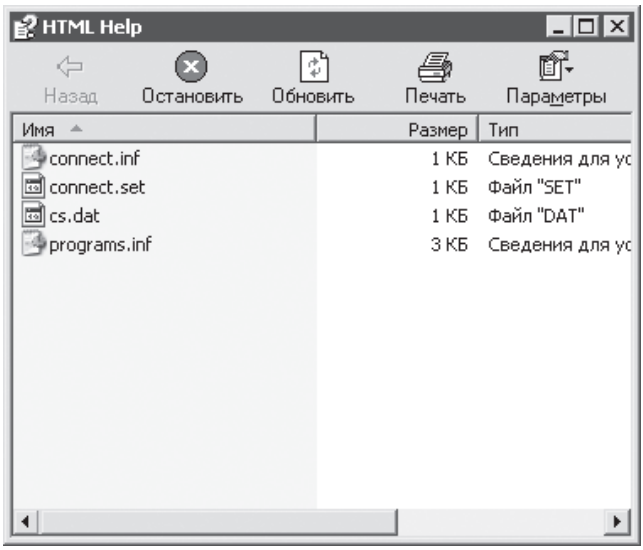


Рис. 1.14. Вызов программы HTML Help

Вот и все стандартные ActiveX-объекты, которые, кроме `DllRegisterServer` и `DllUnregisterServer`, поддерживают дополнительные функции.

Драйверы

Драйверы — это специальные программы, управляющие работой оборудования, для которого они были написаны. Сейчас практически каждое устройство — от оптической мыши до видеокамеры — требует для своего взаимодействия с компьютером отдельный драйвер. Конечно, не все из этих устройств поставляются со своим собственным драйвером — и это неудивительно, ведь операционная система Windows XP уже включает в себя очень много различных драйверов для многих устройств известных производителей. Все они хранятся на установочном диске (а также в локальной файловой системе) в архиве `driver.cab` и устанавливаются только при нахождении системой нового оборудования.

Как и ActiveX-объекты, драйверы также могут хранить вызов функций, но использование этих функций оказывается не всегда возможным. Во-первых, некоторые драйверы написаны для операционной системы MS-DOS, поэтому применение их функций в операционной системе Windows XP невозможно. Во-вторых, вызов функции уже загруженного драйвера не всегда срабатывает — в большинстве случаев система просто выведет сообщение, подобное приведенному на рис. 1.15.

У себя на компьютере автор книги нашел только одну доступную команду, предоставляемую драйвером `WINSPOOL.DRV`. Это стандартный драйвер Windows, расположенный в каталоге `%systemroot%\system32`. Он управляет подключением к принтеру (а также передачей ему заданий) и поддерживает такую команду `rundll32.exe`: `rundll32 WINSPOOL.DRV, ConnectToPrinterDlg`. Она

выводит диалог для подключения к сетевому принтеру и передачи ему задания. Результат выполнения данной команды будет подобным приведенному на рис. 1.16.

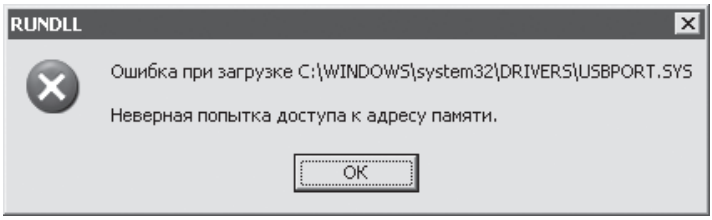


Рис. 1.15. Как правило, уже загруженные драйверы команды не поддерживают

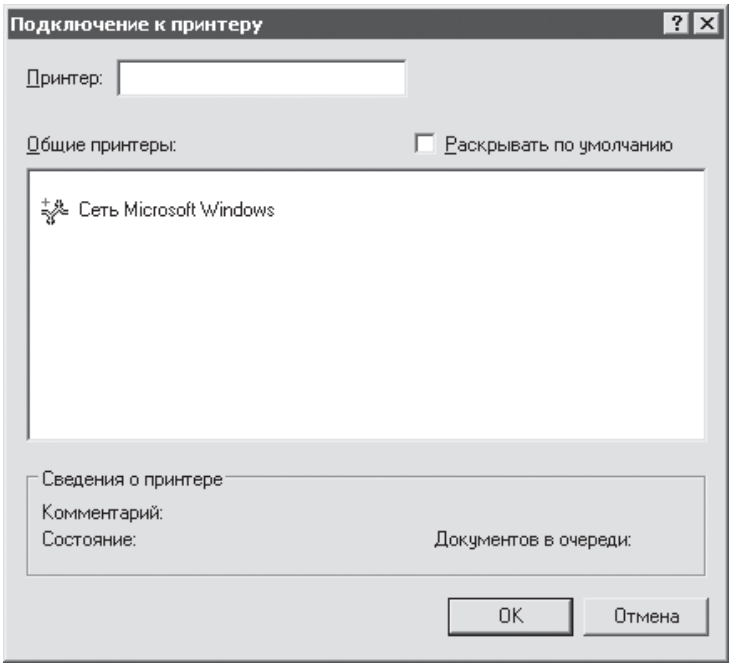


Рис. 1.16. Результат выполнения команды rundll32 WINSPOOL.DRV, ConnectToPrinterDlg

ПРИМЕЧАНИЕ

Это диалоговое окно также можно вызвать с помощью команды rundll32.exe PRINTUI.dll, ConnectToPrinterDlg.

Библиотеки

Библиотеки — это специальные файлы с расширением DLL, описывающие все функции, применяемые программой, для которой данная библиотека написана.

Именно библиотеки содержат большинство функций, используемых операционной системой или программой. При этом библиотеки могут применяться как в программах (даже если данная библиотека не была специально написана для соответствующей программы), так и в драйверах, CPL-файлах, ActiveX-объектах и других исполняемых файлах Windows.

Раньше были рассмотрены команды `rundll32.exe` по файлам, в которых они описаны. Это было более эффективно по той причине, что стандартных CPL-файлов или файлов ActiveX-объектов на компьютере не очень много. Другая картина предстает при взгляде на количество библиотек, зарегистрированных в системе. Чтобы в этом убедиться, достаточно просто заглянуть в каталог `%systemroot%\system32` — именно он содержит стандартные библиотеки Windows, которых в этом каталоге можно насчитать сотни. А ведь с каждой программой поставляются еще и свои собственные библиотеки, которые могут находиться как в каталоге, в котором была установлена программа, так и в `%systemroot%\system32`. Поэтому на этот раз команды `rundll32.exe` будут рассмотрены не по библиотекам, в которых они описаны, а по действиям, которые они выполняют.

Окна работы с сетью

Первым рассмотренным вопросом будет вопрос вызова различных мастеров, предназначенных для установки тех или иных сетевых компонентов компьютера. Автор знает по своему опыту, как иногда бывает сложно найти в файловой системе Windows ссылки на вызовы тех или иных мастеров настройки не только сетевых компонентов, но и любых других частей Windows. Особенно это актуально после преднамеренного удаления данных ссылок на диалоговые окна — когда пользователь хочет минимизации содержимого Windows и удаляет из меню Пуск или из других мест файловой системы все подряд, что ему кажется ненужным.

Рассмотрим команду вызова Мастера установки оборудования. Для его вызова достаточно выполнить команду `rundll32.exe ccfgnt.dll, IcfgInstallModem`, результат действия которой можно увидеть на рис. 1.17.

Аналогичное окно можно вызвать и с помощью следующей команды: `rundll32.exe modemui.dll, InvokeControlPanel`.

Если модем уже настроен, то можно вызвать диалог отображения свойств модемов, являющийся апплетом Телефон и модем (`telephon.cpl`) (рис. 1.18). Для этого применяется команда `rundll32.exe TAPI32.dll, internalConfig`.

С помощью данного диалога можно настроить дополнительные параметры инициализации модема или используемые им драйверы — для этого применяется вкладка Модемы. Но, кроме вкладки Модемы, данный диалог содержит еще две вкладки. С помощью вкладки Набор номера можно настроить параметры доступа к поставщику услуг Интернета, а с помощью вкладки Дополнительно можно добавить, настроить или удалить службы, которые используются при работе модемов.

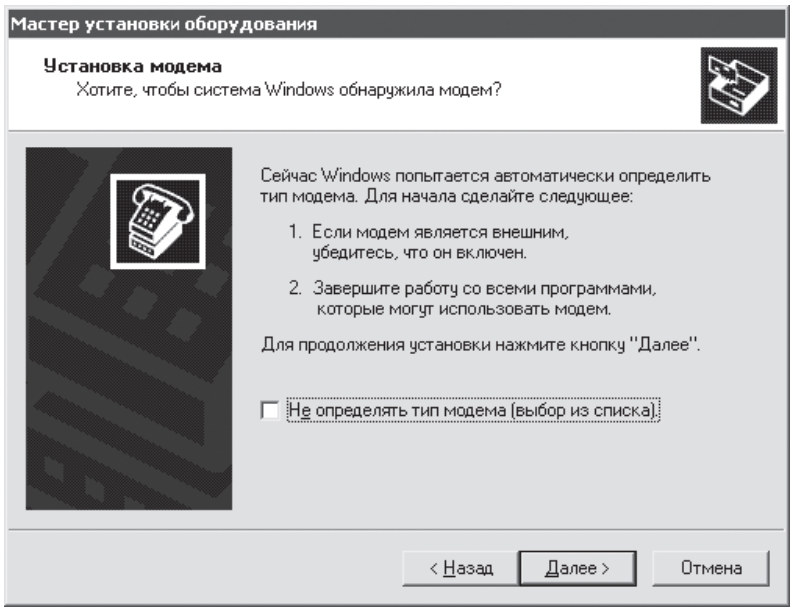


Рис. 1.17. Результат выполнения команды rundll32.exe ccfgnt.dll, lcfgInstallModem

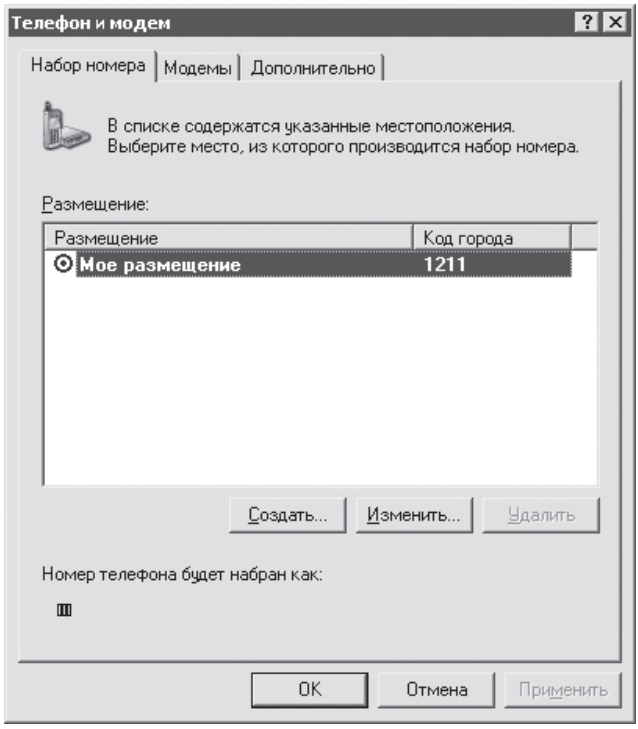


Рис. 1.18. Результат выполнения команды rundll32.exe TAPI32.dll, internalConfig

ПРИМЕЧАНИЕ

Если вам необходима только вкладка Набор номера, то можно воспользоваться командой rundll32.exe TAPI32.dll, LOpenDialAsst. С ее помощью можно вывести приведенное выше диалоговое окно, в котором будет доступна только вкладка Набор номера.

Можно вызвать Мастер настройки сети, предназначенный для создания и настройки сетевого соединения между несколькими компьютерами. Для этого необходимо воспользоваться командой `rundll32.exe hnetwiz.dll, HomeNetWizardRunDll`. Можно также обратиться к другому мастеру для создания сети — Мастеру новых подключений. С его помощью можно подключиться к Интернету, частной сети или сконфигурировать свою сеть. Для вызова этого мастера достаточно воспользоваться командой `rundll32.exe netshell.dll, StartNCW`. Можно также выполнить команду `rundll32.exe RASAPI32.dll, RasCreatePhonebookEntryA`. С ее помощью отображается мастер с названием, аналогичным предыдущему (Мастер новых подключений), но немного другой функциональностью. На рис. 1.19 можно увидеть два этих мастера (справа находится мастер, вызываемый командой `rundll32.exe RASAPI32.dll, RasCreatePhonebookEntryA`, а слева приведено окно второго шага мастера, вызываемого командой `rundll32.exe netshell.dll, StartNCW`).

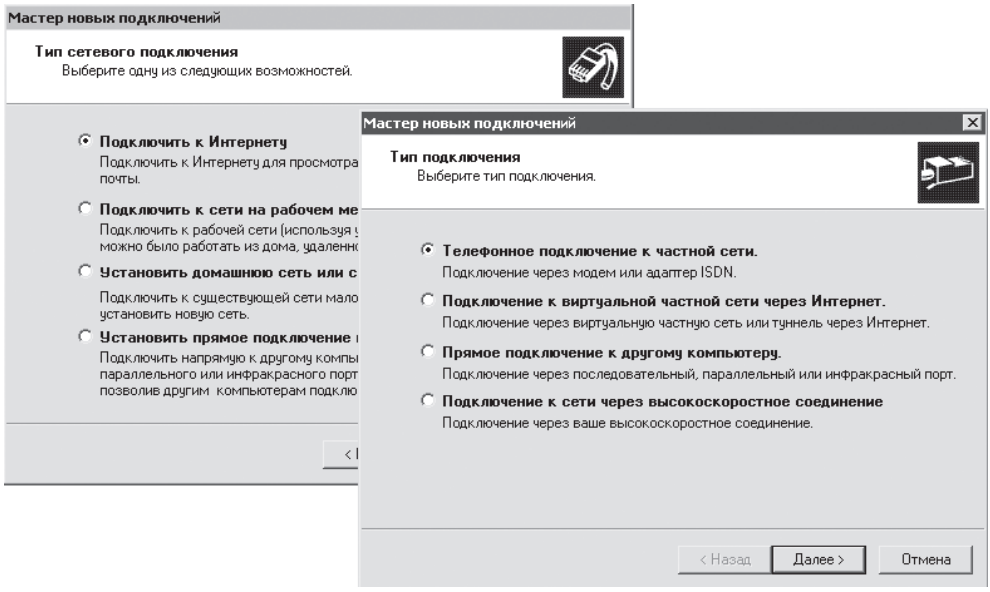


Рис. 1.19. Окна мастера подключения к сети

Но это еще не все, ведь кроме обычных сетей — с помощью подключения нескольких компьютеров к одному кабелю — существуют еще и беспроводные сети. Мастер их создания также можно вызвать. Для этого применяется команда `rundll32.exe wzcdlg.dll, FlashConfigCreateNetwork`. Этот же мастер

можно вызвать и с помощью следующей команды: `rundll32.exe wzcdlg.dll, FlashConfigRunWizard`.

Но если сетевое соединение уже имеется, то можно воспользоваться командой `rundll32.exe netplwiz.dll, AddNetPlaceRunDll` для вызова Мастера добавления в сетевое окружение. С его помощью можно создать ярлык для веб-узла, FTP-узла, удаленного компьютера и т. д.

Еще одной интересной возможностью является вызов диалога установки дополнительных сетевых компонентов (рис. 1.20), с помощью которого можно установить службы печати для UNIX, различные сетевые службы (одноранговую сеть, слушатель RIP и т. д.), а также средства для наблюдения за сетевыми подключениями (например, WMI-поставщик SNMP и сам протокол SNMP). Чтобы вызвать данный диалог, достаточно воспользоваться командой `rundll32.exe netshell.dll, HrLaunchNetworkOptionalComponents`.

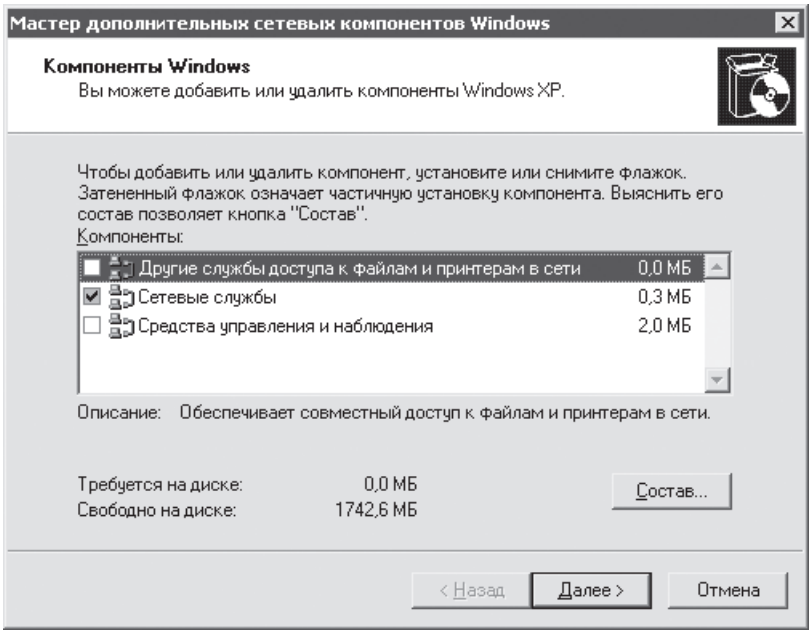


Рис. 1.20. Результат выполнения команды rundll32.exe netshell.dll, HrLaunchNetworkOptionalComponents

Кроме различных мастеров установки или работы с сетью, операционная система Windows предоставляет множество других диалогов, которые косвенно или явно относятся к работе с сетевыми компонентами Windows. К таким диалогам можно отнести диалог Настройка автономных файлов, который является вкладкой Автономные файлы диалога Свойства папки. С его помощью можно определить, будут ли использоваться автономные файлы (файлы удаленного компьютера, доступ к которым можно получить даже тогда, когда соответствующий компьютер отключен), и если будут, то когда будут синхронизироваться автономные файлы. Мож-

но также определить место на диске, отводимое для автономных файлов и т. д. Чтобы вызвать диалог Настройка автономных файлов, достаточно воспользоваться командой `rundll32.exe cscui.dll, CSCOptions_RunDLL`.

Существует возможность вызова диалога Параметры синхронизации, с помощью которого можно определить, какие файлы будут синхронизироваться при входе/выходе или простое системы, а также с помощью какого сетевого подключения они будут синхронизироваться (рис. 1.21). Для этого применяется команда `rundll32.exe mobsync.dll, DisplayOptions`.

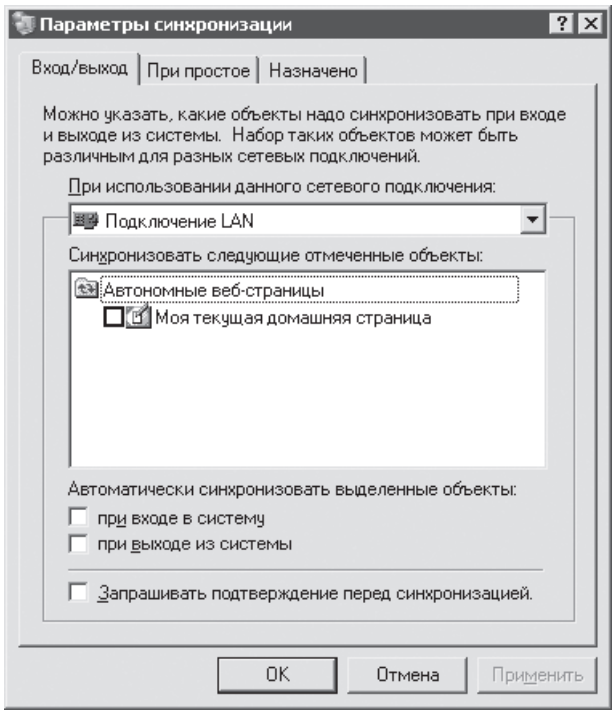


Рис. 1.21. Результат выполнения команды rundll32.exe mobsync.dll, DisplayOptions

Операционная система Windows позволяет создавать не только автономные файлы. В ней существуют папки, общий доступ к которым можно открыть по сети. По умолчанию в Windows существует стандартный набор скрытых папок общего доступа, определяющих логические диски компьютера, а также два скрытых общих ресурса `ADMIN$` и `IPCS$` (знак `$` в конце общего ресурса как раз и говорит о том, что он является скрытым), предназначенных для администрирования компьютера. Кроме стандартных папок общего доступа, Windows позволяет создать свои открытые папки. Для этого можно воспользоваться несколькими способами — либо создать папку с помощью вкладки Доступ диалога Свойства для папки (нельзя создавать папки общего доступа, если отключена служба Сервер), либо с помощью соответствующей оснастки. Если вы выбрали второй способ, то диалог для создания общего ресурса можно вызвать с помощью команды `rundll32.exe`

`ntlanman.dll, ShareCreate`. Его же можно вызвать и с помощью команды `rundll32.exe ntlanui.dll, ShareCreate`, являющейся аналогом предыдущей.

Если же вы выбрали первый способ, то для быстрого доступа к вкладке Доступ диалога Свойства можно воспользоваться следующей командой — `rundll32.exe ntshrui.dll, SharingDialog «ïóòü ê ïàïêå»`. После ее вызова перед вами отобразится довольно экзотический диалог (если посмотреть на расположение кнопок ОК и Отмена), но тем не менее, несмотря на некоторые недостатки, он работает (рис. 1.22).

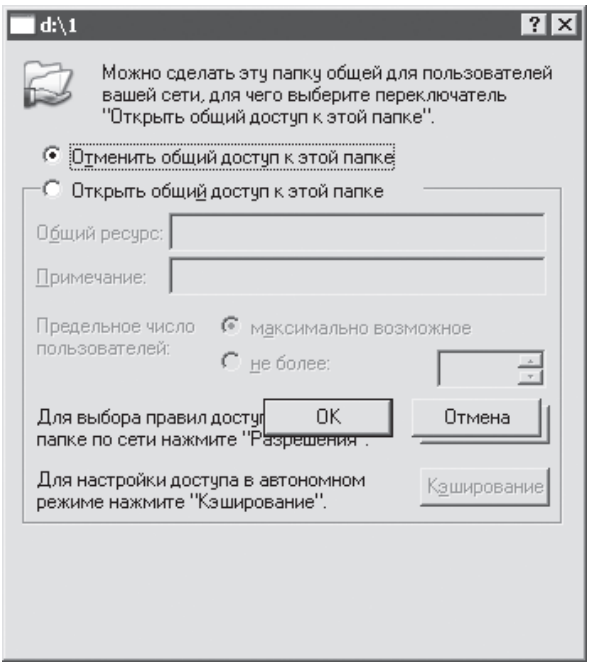


Рис. 1.22. Конечно, диалог с небольшими дефектами, но зато работает

Существует возможность просмотра списка всех общих ресурсов, доступных на данном компьютере. Для этого применяется оснастка Общие папки. Но более быстрым способом является непосредственный вызов диалога Общие папки — для этого достаточно просто воспользоваться командой `rundll32.exe ntlanman.dll, ShareManage`. Кроме просмотра списка общих ресурсов, этот диалог позволяет создать новый общий ресурс, просмотреть свойства общего ресурса или прекратить доступ к нему.

ПРИМЕЧАНИЕ

Диалог можно вызвать и с помощью команды rundll32.exe ntlanui.dll, ShareManage, являющейся аналогом предыдущей команды.

Мастера установки других компонентов Windows

Как можно было заметить, в операционной системе Windows существует очень много мастеров работы с сетью. Это связано с тем, что настройка сети считается сложной задачей, а Microsoft с каждой версией своей операционной системы пытается все больше упростить функции администрирования Windows. Хотя, глядя на такое разнообразие мастеров настройки сети, легко запутаться в том, для чего они применяются и какой из них лучше использовать. Совершенно по-другому обстоят дела с мастерами настройки других компонентов Windows — их не очень много и можно пересчитать по пальцам. Тем не менее они есть, и уже ради этого стоит рассмотреть способы их вызова. Этим мы сейчас и займемся.

- `rundll32.exe sti_ci.dll, AddDevice` — с помощью данной команды можно вызвать Мастер установки сканера или цифровой камеры. То же самое делает команда `rundll32.exe wiashext.dll, AddDeviceWasChosen`.
- `rundll32.exe TCPMonUI.dll, LocalAddPortUI` — позволяет вызвать Мастер добавления стандартного порта TCP/IP принтера (рис. 1.23), с помощью которого можно подключиться к удаленному принтеру.

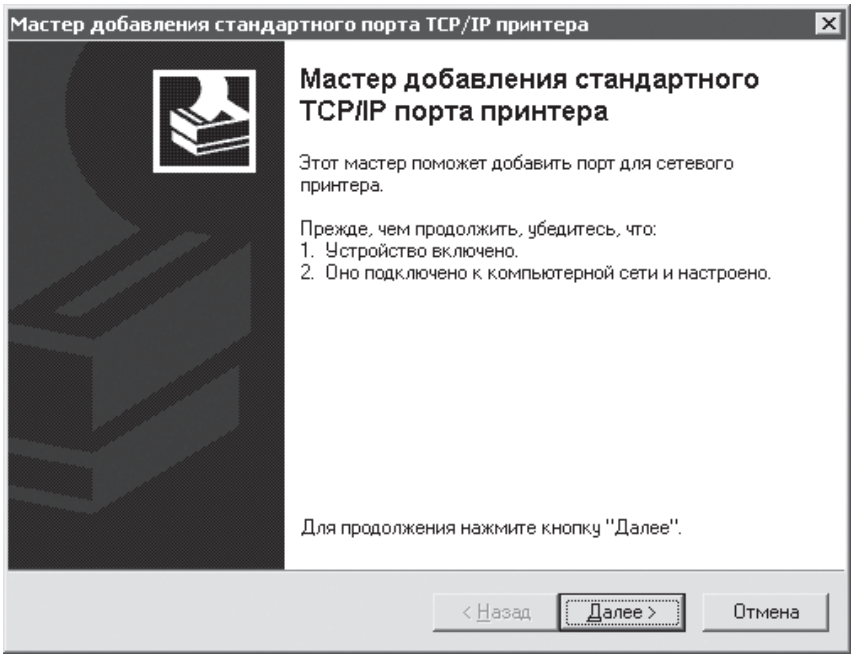


Рис. 1.23. Результат выполнения команды rundll32.exe TCPMonUI.dll, LocalAddPortUI

- `rundll32.exe upnpui.dll, InstallUPnPUI` — при вызове данной команды в сетевом окружении появляются значки сетевых UPnP-устройств. Существует также команда и для удаления сетевых UPnP-устройств — `rundll32.exe upnpui.dll, UnInstallUPnPUI`.

Другие мастера

Теперь поговорим о различных мастерах, не относящихся к настройке сети или каких-нибудь компонентов Windows. Например, к таким мастерам можно отнести Мастер сброса паролей (рис. 1.24). С его помощью можно сбросить текущий пароль учетной записи пользователя, предоставив данному мастеру дискету с информацией о пароле. Чтобы вызвать окно данного мастера, достаточно воспользоваться командой `rundll32.exe KEYMGR.dll, PRShowRestoreFromMsginaW`. Можно также применить следующую команду: `rundll32.exe KEYMGR.dll, PRShowRestoreWizardExW`. Она выполняет аналогичные действия.

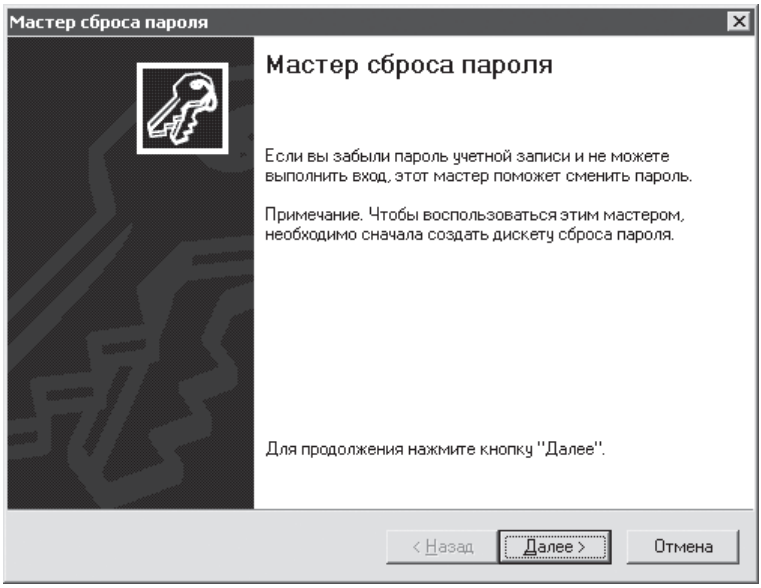


Рис. 1.24. Результат выполнения команды rundll32.exe KEYMGR.dll, PRShowRestoreFromMsginaW

Конечно, необходимость данного мастера немного спорна, ведь если вы вошли в систему, то и так знаете пароль учетной записи данного пользователя и сбросить его можно и без предоставления дискеты с информацией о пароле. Кстати, если у вас нет дискеты с паролем, то данным мастером будет довольно сложно воспользоваться. Поэтому предлагаю сейчас же ее создать. Для этого применяется диалог Мастер забытых паролей (рис. 1.25), отобразить который можно с помощью команды `rundll32.exe KEYMGR.dll, PRShowSaveFromMsginaW`. Аналогичные действия выполняет следующая команда: `rundll32.exe KEYMGR.dll, PRShowSaveWizardExW`.

Следующий мастер, который можно вызвать, — Мастер паспорта .NET (рис. 1.26). Такой паспорт может требоваться при посещении некоторых сайтов Интернета (например, сайта почтовой службы Hotmail). Чтобы вызвать данный мастер, достаточно воспользоваться командой `rundll32.exe NETPLWIZ.dll, PassportWizardRunDll`. Паспорт .NET позволяет использовать один и тот же логин и пароль на всех

сайтах, которые поддерживают работу с ним. В паспорте также можно указать дополнительную информацию, тогда ее не придется вводить в формах сайтов. Программисты Microsoft утверждают, что для защиты вашего паспорта .NET используется мощная система шифрования, поэтому взломать его практически невозможно.

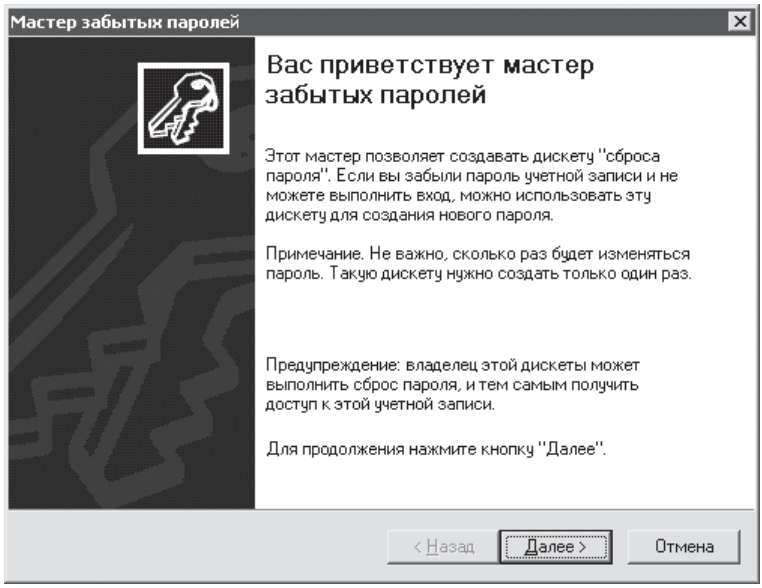


Рис. 1.25. Результат выполнения команды rundll32.exe KEYMGR.dll, PRShowSaveFromMsginaW

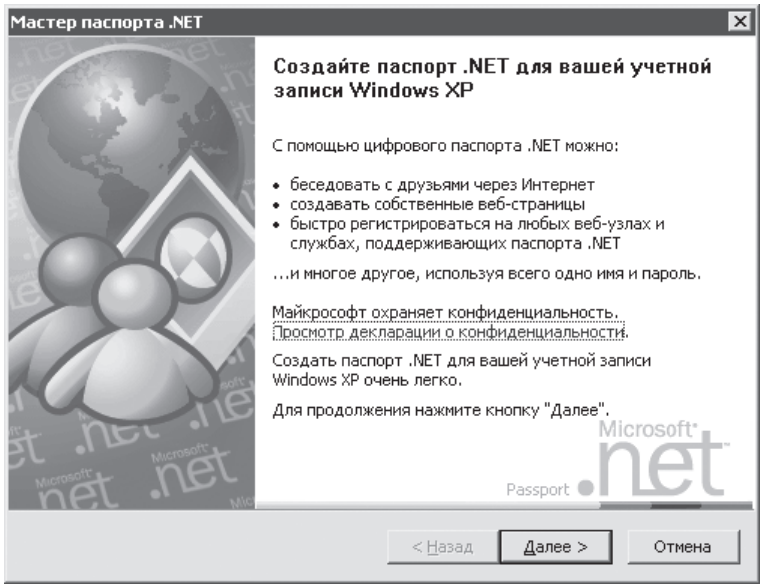


Рис. 1.26. Результат выполнения команды rundll32.exe NETPLWIZ.dll, PassportWizardRunDll

Можно вызвать Мастер веб-публикаций, с помощью которого можно опубликовать различные файлы в Интернете или в локальной сети. После этого данные файлы можно будет просмотреть с помощью браузера. Чтобы вызвать данный мастер, достаточно выполнить команду `rundll32.exe NETPLWIZ.dll, PublishRunDll`.

Диалоги завершения работы Windows

Еще одной разновидностью диалогов, которые используются в Windows, являются различные диалоги-предупреждения, применяемые при попытке выхода пользователя из системы или при необходимости перезагрузки. Большая их часть является обычными окнами, не несущими никакой функциональности, хотя попадаются и такие, которые действительно перезагружают компьютер.

Одним из таких «полнофункциональных» диалогов является окно Мастера настройки Internet Explorer 6 (рис. 1.27), отображаемое при выполнении команды `rundll32.exe IEAKENG.dll, DoReboot`. После нажатия кнопки Да компьютер действительно начнет перезагружаться.

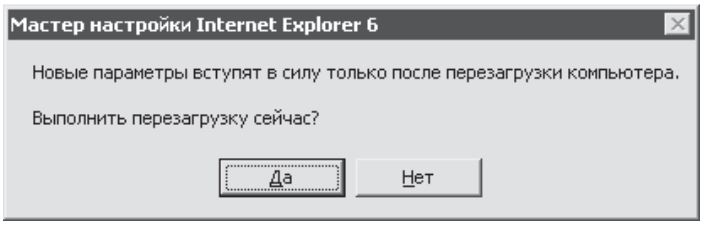


Рис. 1.27. Результат выполнения команды rundll32.exe IEAKENG.dll, DoReboot

Существует возможность перезагрузить компьютер без вывода каких-либо диалогов. Для этого применяется следующая команда: `rundll32.exe IUENGINE.dll, EngRebootMachine`, используемая службой Windows Update. После ее выполнения система автоматически закроет все работающие приложения и начнет перезагрузку компьютера.

Вот, в принципе, и все команды `rundll32.exe`, которые действительно перезагружают компьютер. Но, кроме них, существует еще несколько интересных команд, которые не перезагружают компьютер, но угрожают это сделать. К таким командам можно отнести `rundll32.exe MSGINA.dll, ShellShutdownDialog`, вызывающую диалоговое окно завершения работы компьютера, в списке которого доступны команды перезагрузки, выключения и перехода в спящий режим (рис. 1.28).

С помощью команды `rundll32.exe SHELL32.dll, RestartDialogEx` можно вызвать диалоговое окно Изменение параметров системы, в котором говорится, что после произведенных изменений системы необходимо перезагрузить компьютер.

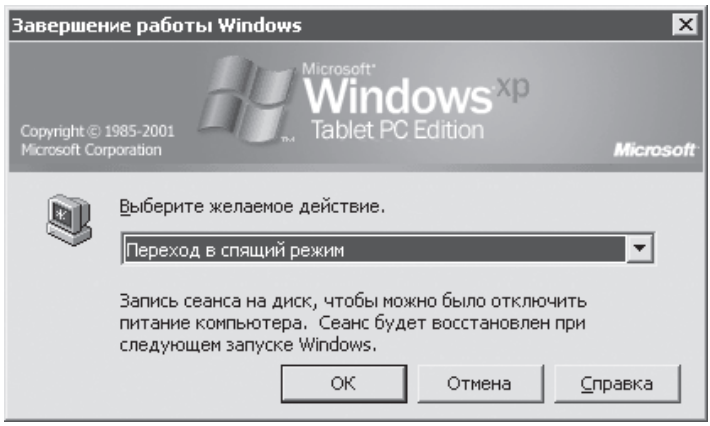


Рис. 1.28. Результат выполнения команды rundll32.exe MSGINA.dll, ShellShutdownDialog

Еще одним «нефункциональным» диалоговым окном, которое можно вызвать, является окно с сообщением Данный компьютер используется другим пользователем. Для вызова этого окна необходимо выполнить команду `rundll32.exe USER32.dll, DisplayExitWindowsWarnings`.

Другие диалоговые окна

В Windows XP присутствуют не только диалоговые окна и мастера для работы с сетью — есть и много других диалоговых окон, некоторые стоят того, чтобы их рассмотреть. Например, существует возможность открытия диалогового окна Сертификаты, предназначенного для просмотра сведений о различных типах сертификатов, установленных на компьютере. Для этого можно выполнить команду `rundll32.exe CRYPTUI.dll, CryptUIStartCertMgr`, результат чего представлен на рис. 1.29.

Для открытия диалога Сертификаты можно применить еще несколько команд. Первой является команда `rundll32.exe IEAKENG.dll, ModifySiteCert`. Вызов ее эквивалентен вызову предыдущей команды. Этот же диалог можно вызвать и с помощью следующей команды: `rundll32.exe wintrust.dll, OpenPersonalTrustDBDialog`.

Существует возможность вызова диалога Сертификаты, открытого на вкладке Доверенные издатели (в диалоге будет присутствовать только эта вкладка). Для этого необходимо воспользоваться командой `rundll32.exe IEAKENG.dll, ModifyAuthCode`.

Кроме диалога Сертификаты, команды `rundll32.exe` позволяют вызвать еще и окно Диспетчера устройств. Для этого достаточно воспользоваться командой `rundll32.exe devmgr.dll, DeviceManager_Execute`. Это же действие можно выполнить с помощью команды `rundll32.exe devmgr.dll, DevicePropertiesA`.

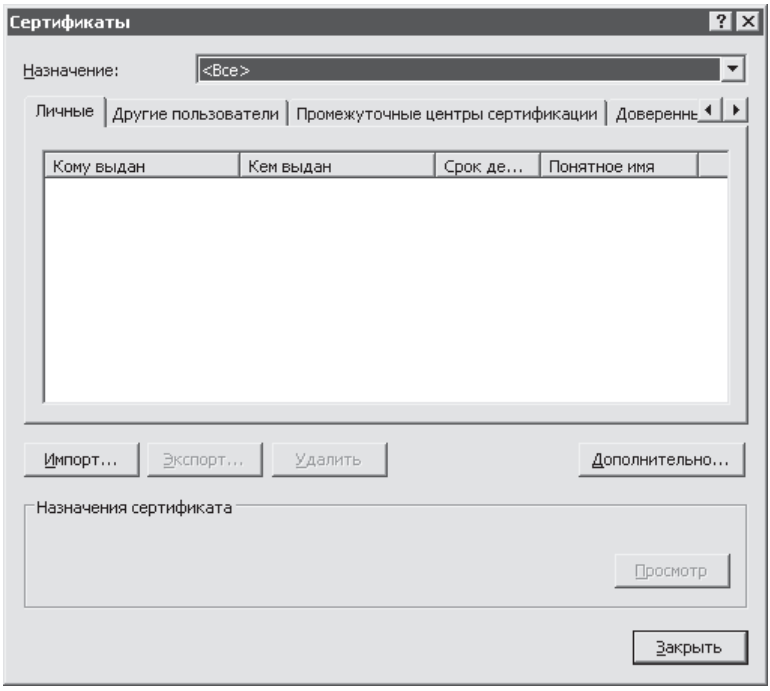


Рис. 1.29. Результат выполнения команды rundll32.exe CRYPTUI.dll, CryptUIStartCertMgr

Но и это еще не все. Операционная система Windows XP позволяет открывать следующие диалоговые окна.

- `Rundll32.exe diskcopy, DiskCopyRunDll` — позволяет открыть диалоговое окно копирования содержимого одной дискеты на другую (рис. 1.30). После нажатия кнопки Начать будет произведено копирование в память компьютера содержимого дискеты, установленной в дисковод A:. После этого система попросит вставить новую дискету в дисковод A:, на которую будут скопированы данные из памяти компьютера.

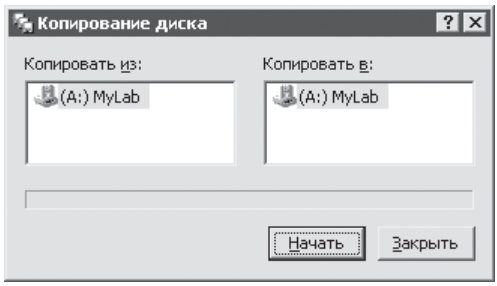


Рис. 1.30. Результат выполнения команды Rundll32.exe diskcopy, DiskCopyRunDll

- `rundll32.exe dsquery.dll, OpenQueryWindow` — вызов данной команды открывает диалог для поиска в каталоге Active Directory. Если у вас не ус-

тановлен Active Directory, то диалог все равно будет вызываться, но пользоваться им будет нельзя.

- `rundll32.exe FldrClnr.dll, Wizard_RunDLL ALL` — позволяет вызвать Мастер очистки рабочего стола, с помощью которого можно перенести неиспользуемые ярлыки Рабочего стола в специальную папку Неиспользуемые ярлыки. Доступ к этому мастеру также можно получить, нажав кнопку Очистить рабочий стол на вкладке Общие диалога Элементы рабочего стола. Диалог вызывается нажатием кнопки Настройка рабочего стола на вкладке Рабочий стол окна Свойства: Экран.
- `rundll32.exe IEAKENG.dll, BrowseForFolderA «òåêñò»` — вызов данной команды отображает диалог, подобный представленному на рис. 1.31. Окно используется для выбора папки, но в нашем случае выбранную папку будет некуда передать, поэтому выполнение данной команды в конечном итоге будет приводить к ошибке. Этот диалог можно вызвать и с помощью команды `rundll32.exe IUENGINE.dll, EngBrowseForFolder`.

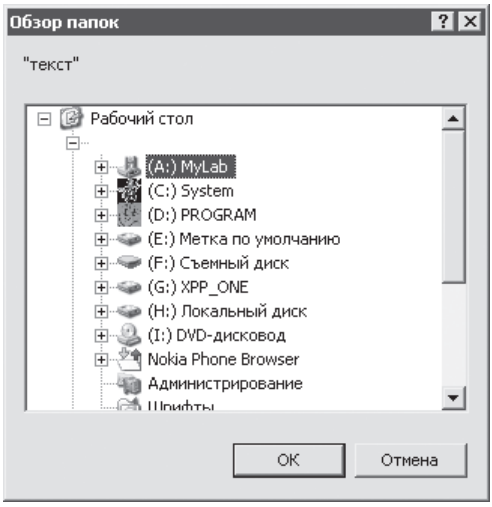


Рис. 1.31. Результат выполнения команды rundll32.exe IEAKENG.dll, BrowseForFolderA «текст»

- `rundll32.exe IEAKENG.dll, ShowDeskCpl` — позволяет вывести диалог Свойства: Экран, в котором будет присутствовать только одна вкладка — Рабочий стол. Все остальные вкладки будут скрыты с помощью групповых политик при запуске данной команды (если пользователь не имеет прав на изменение параметров групповых политик, то отображаемый диалог будет содержать все вкладки). После закрытия диалогового окна Свойства: Экран все настройки групповых политик, которые были изменены данной командой, будут удалены.
- `rundll32.exe KEYMGR.dll, KRShowKeyMgr` — вызов данной команды отображает диалоговое окно Сохранение имен пользователей и паролей (рис. 1.32). С помощью этого диалога можно указать или удалить имена пользователей и пароли к различным серверам домена или сети Интернет. После указания этих паролей операционная система Windows будет автоматически их использовать, не спрашивая у вас пароль.

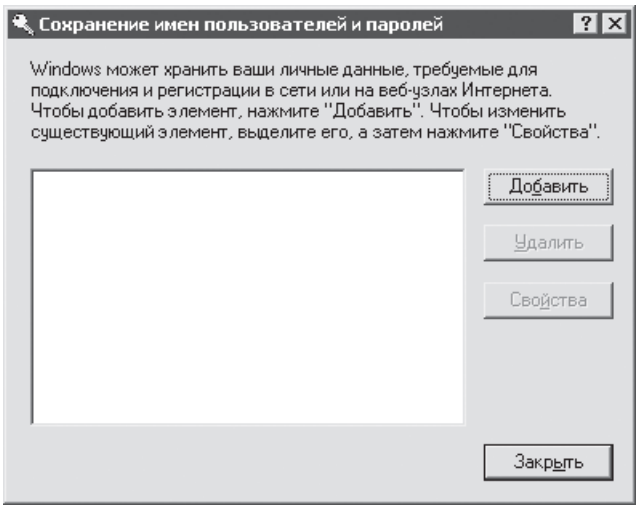


Рис. 1.32. Результат выполнения команды rundll32.exe KEYMGR.dll, KRShowKeyMgr

- `rundll32.exe MSCTF.dll, TF_RunInputCPL` — с помощью данной команды вызывается диалог Языки и службы текстового ввода. Благодаря ему можно определить конфигурации клавиш для смены раскладки клавиатуры, раскладку клавиатуры по умолчанию, а также определить настройки языковой панели и дополнительных служб текстового ввода. Доступ к этому диалогу можно получить и после нажатия кнопки Подробнее на вкладке Языки окна Язык и региональные стандарты (`intl.cpl`).
- `rundll32.exe netplwiz.dll, UsersRunDll` — вызов данной команды отображает диалоговое окно для редактирования списка учетных записей пользователей, зарегистрированных в системе (рис. 1.33). С помощью данного диалога можно указать учетную запись пользователя, с правами которого всегда будет выполняться автоматический вход в систему. Для этого достаточно просто снять флажок Требовать ввод имени пользователя и пароля. После этого система попросит вас ввести логин пользователя, с правами которого будет выполняться вход в систему, и его пароль.

ПРИМЕЧАНИЕ

Возможность автоматического входа в систему с правами конкретного пользователя можно установить с помощью реестра. Для этого применяются следующие параметры строкового типа из ветви реестра HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon: AutoAdminLogon — если значение данного параметра равно 1, то автоматический вход в систему с правами конкретного пользователя будет включен; DefaultDomainName — параметр определяет домен, к которому будет выполняться подключение; DefaultUserName — параметр определяет логин пользователя, от имени которого будет выполняться автоматический вход в систему; DefaultPassword — параметр определяет пароль для учетной записи, от имени которой будет выполняться вход в систему (внимание, этот пароль не шифруется, поэтому любой сможет его узнать, просмотрев данную ветвь реестра).

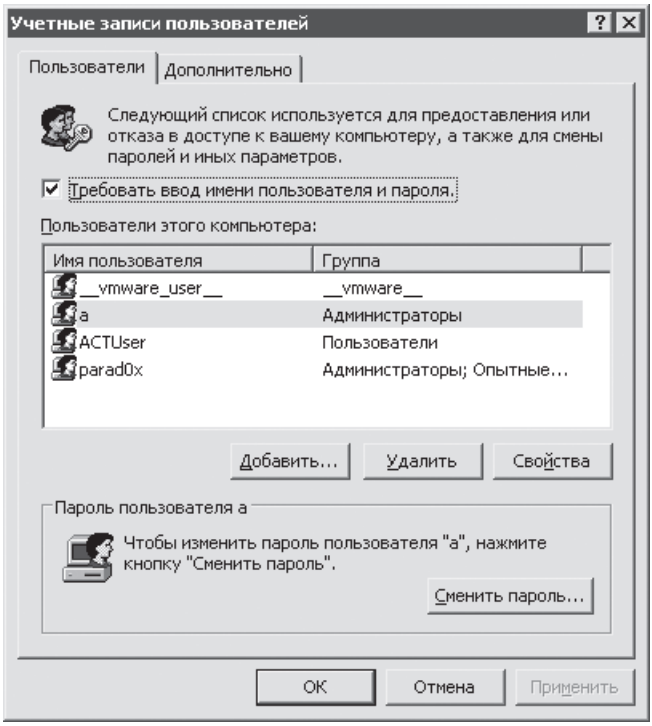


Рис. 1.33. Результат выполнения команды rundll32.exe netplwiz.dll, UsersRunDll

- `rundll32.exe newdev.dll, WindowsUpdateDriverSearchingPolicyUi` — позволяет вызвать диалог подключения к Windows Update (рис. 1.34), с помощью которого можно указать, разрешено ли подключение компьютера к сайту Microsoft для скачивания обновлений операционной системы. Данный диалог также можно вызвать, нажав кнопку Узел Windows Update на вкладке Оборудование диалога Свойства системы.

  Windows Update является простейшим средством для скачивания с узла Microsoft заплат и обновлений для операционной системы Windows или драйверов для различных устройств. Для своей работы сервер Windows Update использует 443 порт, поэтому он должен быть открыт. Перед тем как загрузить на ваш компьютер обновления, Windows Update проверяет версии установленных в операционной системе компонентов, языковые настройки, а также параметры реестра, описывающие лицензирование продукта; если версия операционной системы не является лицензионной (так называемая пиратская версия), то вам будет отказано в доступе к функции Windows Update. Вместе со скачиваемым обновлением будет загружен уникальный идентификатор, применяемый для идентификации вашего компьютера на сервере Windows Update. Корпорация Microsoft утверждает, что этот идентификатор не используется для вашей идентификации в сети Интернет.

- `rundll32.exe RASDLG.dll, RasAutodialDisableDlgA` — вызов данной команды отображает диалог сбоя при автоматическом наборе номера (рис. 1.35).

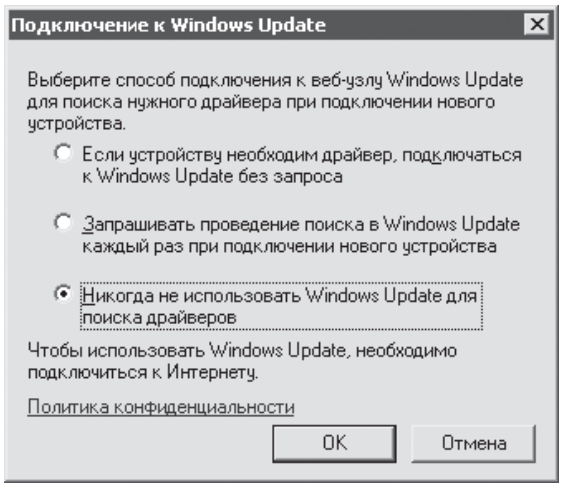


Рис. 1.34. Результат выполнения команды rundll32.exe newdev.dll, WindowsUpdateDriverSearchingPolicyUi

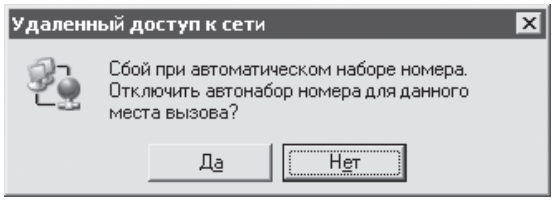


Рис. 1.35. Результат выполнения команды rundll32.exe RASDLG.dll, RasAutodialDisableDlgA

Если нажать кнопку Да этого диалога, то DWORD-параметру 1 будет присвоено значение 1. Параметр расположен в ветви реестра HKEY_CURRENT_USER\Software\Microsoft\RAS AutoDial\Control\Locations.

- `rundll32.exe shdocvw.dll, DoOrganizeFavDlg` — позволяет вызвать диалог Упорядочить избранное, отображенный на рис. 1.36. С его помощью можно создавать, удалять или перемещать папки.
- `rundll32.exe SHELL32.dll, Options_RunDLL 0` — вызов данной команды отображает диалоговое окно Свойства папки, с помощью которого можно изменить некоторые стандартные настройки оболочки пользователя, отредактировать параметры различных типов файлов, а также получить доступ к вкладке Автономные файлы. Данный диалог можно открыть, выбрав в меню Сервис пункт Свойства папки.

  Возможно использование и другой разновидности данной команды, а именно `rundll32.exe SHELL32.dll, Options_RunDLL 1`. Она позволяет открыть диалоговое окно Свойства панели задач и меню Пуск. Его же можно открыть с помощью команды Свойства контекстного меню Панели задач.
- `rundll32.exe shell32.dll, SHFormatDrive` — выполнение данной команды приводит к открытию диалога форматирования (рис. 1.37).

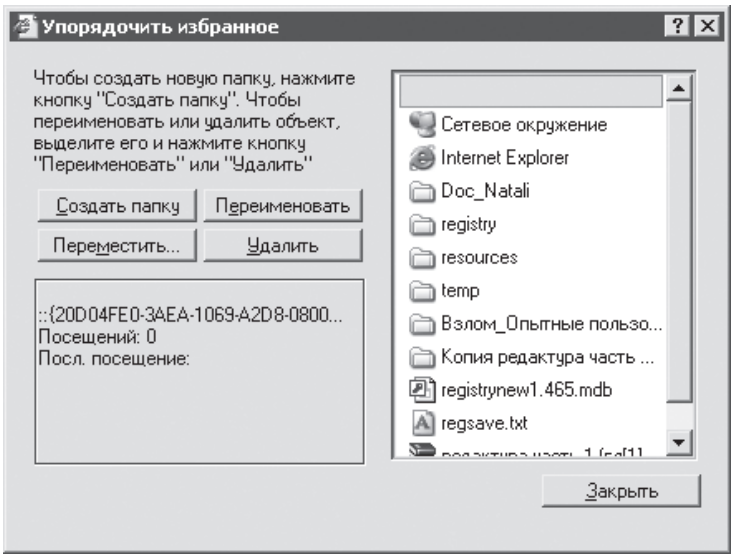


Рис. 1.36. Результат выполнения команды rundll32.exe shdocvw.dll, DoOrganizeFavDlg

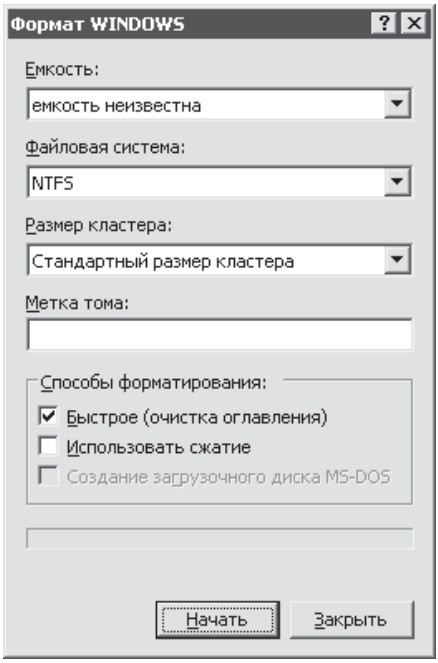


Рис. 1.37. Результат выполнения команды rundll32.exe shell32.dll, SHFormatDrive

- `rundll32.exe syncui.dll, Briefcase_Intro` — с помощью данной команды можно отобразить диалог, появляющийся при создании папки Портфель на Рабочем столе с помощью функций оболочки Windows (рис. 1.38).

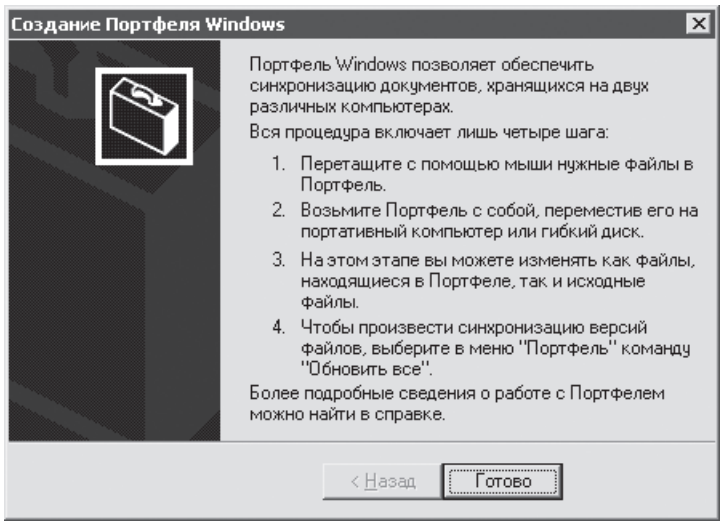


Рис. 1.38. Результат выполнения команды rundll32.exe syncui.dll, Briefcase_Intro

При этом сама папка Портфель на Рабочем столе создана не будет — для этого необходимо использовать следующую команду: `rundll32.exe syncui.dll, Briefcase_Create`.

Глава 2

Конфигурация

- **Конфигурация Windows**
- **Файловая система**
- **Другие операции**

Программа `rundll32.exe` применяется не только для вызова многообразных диалоговых окон Windows. С ее помощью можно выполнить конфигурирование различных настроек операционной системы Windows — от переустановки всевозможных компонентов Windows до восстановления стандартных параметров пользовательской настройки самой операционной системы. Вкратце опишем команды для решения этих задач.

Перед рассмотрением хотелось бы дать небольшой совет — прежде чем выполнить любую из приведенных ниже функций, особенно если эти функции лишают операционную систему какого-нибудь ее составляющего, нужно два и даже три раза подумать. Ведь никто не может дать гарантии, что после выполнения той или иной команды ваша операционная система сможет загрузиться или работать стабильно.

Конфигурация Windows

Первым рассмотренным вопросом будет возможность удаления либо повторной регистрации тех или иных частей операционной системы. Это может быть полезно не только при ограничении доступа к различным компонентам, но и для восстановления поврежденных частей операционной системы Windows XP.

Конфигурация компонентов

Сначала будут описаны вопросы работы с некоторыми компонентами системы с помощью команд `rundll32.exe`. Не будем углубляться в функционирование данных компонентов, ведь книга, которую вы держите в руках, написана совершенно не для этого. Вместо этого приведем краткий список команд, с помощью которых можно удалить или заново зарегистрировать в реестре те или иные части операционной системы Windows.

- `AutoDisc.dll` — применяется для повторной регистрации (или удаления) в реестре информации, необходимой для автообнаружения электронной почты почтовым клиентом Outlook Express.
- `btpanui.dll` — используется для повторной регистрации (или удаления) в реестре ActiveX-объектов, необходимых для работы пользовательского интерфейса Bluetooth (Bluetooth PAN User Interface).
- `CABVIEW.dll` — удаление регистрационных данных библиотеки приводит к удалению возможности работы со стандартной программой Windows, предназначенной для работы с файлами с расширением CAB. Регистрация же данной библиотеки приводит к регистрации в реестре расширения CAB.

ПРИМЕЧАНИЕ

Перед удалением регистрационных параметров программы для работы с CAB-файлами операционная система ищет в ветви реестра HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall разделы CabView и MSCabFileView. Если она их найдет, то будут выполнены команды, записанные в этих разделах, а потом эти разделы будут удалены.

Можно также удалить саму программу для чтения CAB-файлов. Для этого используется команда `rundll32.exe CABVIEW.dll, Uninstall`. После ее выполнения операционная система попросит подтвердить удаление программы для просмотра CAB-файлов.

- `camocx.DLL` — применяется для повторной регистрации (или удаления) в реестре информации, необходимой для работы с камерами и сканерами на данном компьютере.
- `capesnpn.dll` — используется для повторной регистрации (или удаления) в реестре информации о расширениях центра сертификации.
- `CdfView.dll` — регистрация данной библиотеки приводит к замене сведений о расширениях файлов, предназначенных для работы с файлами каналов Интернета, стандартными настройками этих расширений.
- `CSCUI.dll` — при рассказе о командах `rundll32.exe`, предназначенных для работы с оболочкой Windows, упоминалось о команде, которая отображает диалог Настройка автономных файлов. Возможность работы с автономными файлами присутствовала еще в Windows 2000, но иногда она может быть лишней или даже совершенно ненужной, например, когда пользователи сети постоянно создают такие автономные файлы, а администратору приходится их удалять, так как необходимости в них нет. В этом случае лучшим решением будет простое удаление всех регистрационных данных о библиотеке `CSCUI.dll`, которая как раз и отвечает за автономные файлы. Для этого, как уже известно, достаточно воспользоваться командой `rundll32.exe CSCUI.dll, DllUnregisterServer`. После этого вкладка Автономные файлы исчезнет из диалогового окна Свойства папки. Для ее возвращения достаточно просто зарегистрировать библиотеку `CSCUI.dll`.

  Другим способом отключения автономных файлов является удаление возможности синхронизации компьютеров, хотя в этом случае будет трудно выполнить синхронизацию компьютера с другими устройствами, например с мобильным телефоном. Чтобы удалить возможность синхронизации, достаточно воспользоваться командой `rundll32.exe mobsync.dll, DllUnregisterServer`.
- `DATACLEN.dll` — позволяет зарегистрировать в системе сведения очистки дисков Windows.
- `DSKQUOTA.dll` — еще одной возможностью, которую предоставляет система, является возможность добавления или удаления настройки квот на диски для различных учетных записей пользователей компьютера. Квоты позволяют указать, сколько места отведено на конкретном диске (диск должен быть отформатирован с помощью файловой системы NTFS) для файлов конкретного пользователя компьютера. Если пользователь попытается занять больше места на диске, чем ему разрешено, то система откажет ему в записи информации на диск (или выведет предупреждение в зависимости от настроек квот). Настройки квот выполняются на вкладке Квота диалога Свойства диска, для которого необходимо указать квоты. Если же диалог Свойства не имеет вкладки Квота, то либо данный диск отформатирован с использованием файловой системы FAT32 или более ранней файловой системы, либо в реестре данные о настройке квот повреждены. Чтобы восстановить их, достаточно воспользоваться

командой `rundll32.exe DSKQUOTA.dll, DllRegisterServer`. Как и предыдущие библиотеки, она также позволяет использовать для удаления возможности создания квот на диски функцию `DllRegisterServer`.

- `dsquery.dll` — описывает возможность поиска в службе каталогов Active Directory и позволяет зарегистрировать или удалить данную возможность.
- `DSSENH.dll` — позволяет зарегистрировать или удалить возможность шифрования по алгоритму Диффи—Хеллмана. Можно также зарегистрировать алгоритм шифрования для смарт-карт Gemplus. Для этого используется библиотека `gpkcsp.dll`.

ПРИМЕЧАНИЕ

Алгоритм Диффи—Хеллмана (в Windows он также называется протоколом распределения открытых ключей) позволяет двум участникам сетевого соединения совместно создать секретный ключ, который будет использоваться ими для шифрования пакетов данных при передаче по сети. Несмотря на то, что в действительности алгоритм Диффи—Хеллмана не является асимметричным алгоритмом шифрования, он имеет все положительные стороны, присущие таким алгоритмам, поэтому его принято считать именно ассиметричным алгоритмом шифрования.

- `fontext.dll` — перерегистрировать и пересоздать стандартную папку Windows Шрифты (`%systemroot%\Øðèôòû`).
- `INITPKI.dll` — с помощью регистрации данной библиотеки можно переустановить все настройки центров сертификации и PKI.
- `rundll32.exe ncxpnt.dll, InstallSharing` — позволяет переустановить службы работы с папками общего доступа. Если по какой-либо причине на компьютере нельзя создать папки общего доступа (когда отключена служба Сервер, общие папки также создавать нельзя), то можно попробовать воспользоваться данной командой, возможно, были повреждены какие-нибудь сведения реестра, которые можно переустановить.
- `rundll32.exe NTPRINT.dll, ServerInstallW` — позволяет установить диспетчер очереди печати.
- `rundll32.exe shimgvw.DLL, DllRegisterServer` — с помощью данной команды можно зарегистрировать программу просмотра изображений и факсов (после этого в контекстном меню файлов изображений появится команда Просмотр для открытия данных файлов в этой программе). Если же вы не пользуетесь этой программой, то существует возможность ее удаления (при этом произойдет также удаление команды Просмотр контекстного меню файлов изображений). Для этого достаточно выполнить команду `rundll32.exe shimgvw.DLL, DllUnregisterServer`.
- `SlayerXP.DLL` — если в диалоге Свойства исполняемых файлов (или ярлыков исполняемых файлов) на вашем компьютере отсутствует вкладка Совмес-

тимость, с помощью которой можно запустить программу в режиме совместимости с другими операционными системами Windows (рис. 2.1), то можно попробовать зарегистрировать данную библиотеку — именно она отвечает за возможность использования функции совместимости. Можно также воспользоваться функцией `DllUnregisterServer` для скрытия вкладки Совместимость.

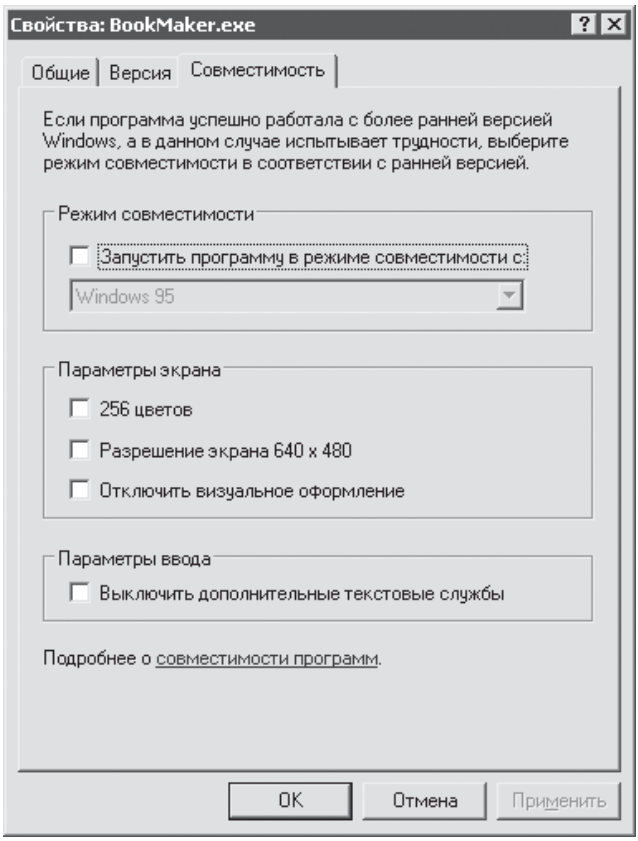


Рис. 2.1. Вкладка Совместимость диалога Свойства исполняемых файлов или ярлыков на них

- `rundll32.exe sti_ci.dll, InstallWiaService` — с помощью данной команды можно установить службу неподвижных изображений. Служба предназначена для работы со сканерами и цифровыми камерами.
- `rundll32.exe WebCheck.dll, DllRegisterServer` — позволяет зарегистрировать возможность подписки веб-узлов.
- `rundll32.exe WININET.dll, DllInstall` — вызов данной команды восстанавливает по умолчанию настройки из ветви реестра `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\Passport`.

Конфигурация стандартных консолей

Консоль — это специальный файл для работы с консолью управления Microsoft (программа `mmc.exe`, о которой будет рассказано в главе 3 книги), имеющий расширение MSC и содержащий одну или несколько оснасток. Кроме пользовательских консолей, которые вы можете создать на основе оснасток, используемых вами в работе чаще всего, существуют также стандартные консоли, поставляемые вместе с операционной системой Windows. Вот о работе этих стандартных консолей мы сейчас вкратце и поговорим.

Большая часть стандартных консолей Windows включает в себя одну оснастку. Каждая оснастка должна быть зарегистрирована в реестре, чтобы ею можно было воспользоваться. Например, если оснастка не будет зарегистрирована в системе, то при попытке открытия консоли, содержащей эту оснастку, перед вами отобразится диалог, подобный приведенному на рис. 2.2.

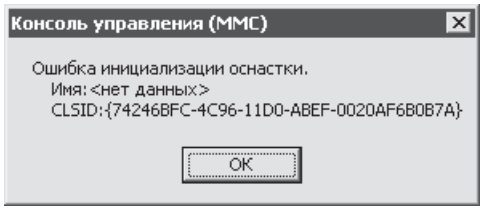


Рис. 2.2. Ошибка инициализации оснастки, вызванная отсутствием сведений о ней в реестре

Если вы когда-нибудь увидите подобный диалог, то не нужно сразу переустанавливать Windows, — все дело в том, что в реестре сведения о данной оснастке были повреждены. Восстановить их можно будет с помощью одной из приведенных ниже команд `rundll32.exe`.

- `rundll32.exe devmgr.dll, DllRegisterServer` — позволяет зарегистрировать оснастку Диспетчер устройств (`devmgmt.msc`), с помощью которой можно просмотреть конфигурацию и состояние установленного на компьютере оборудования.
- `rundll32.exe els.dll, DllRegisterServer` — дает возможность зарегистрировать оснастку Просмотр событий (`eventvwr.msc`), с помощью которой можно просмотреть записи журналов Система, Приложений или Безопасность.
- `rundll32.exe FILEMGMT.DLL, DllRegisterServer` — позволяет зарегистрировать оснастку Общие папки (`fsmgmt.msc`), с помощью которой можно просмотреть сведения обо всех папках общего доступа данного компьютера, а также удалить или добавить общую папку.
- `rundll32.exe GPEDIT.DLL, DllRegisterServer` — дает возможность зарегистрировать оснастку Групповые политики (`gpedit.msc`), с помощью которой настраиваются различные ограничения для учетной записи пользователя.

- `rundll32.exe IEAKSIE.DLL, DllRegisterServer` — с помощью данной команды не регистрируется оснастка как таковая. В данном случае можно зарегистрировать возможность настройки групповых политик для браузера Internet Explorer (доступ к этим настройкам можно получить с помощью оснастки `gpedit.msc` — Конфигурация пользователя ▸ Конфигурация Windows ▸ Настройка Internet Explorer).
- `rundll32.exe IPSECSNP.DLL, DllRegisterServer` — выполнение команды приводит к регистрации в реестре оснастки IPSEC.
- `rundll32.exe IPSMSNAP.DLL, DllRegisterServer` — позволяет зарегистрировать в реестре оснастку Монитор IP-безопасности, с помощью которой можно просмотреть такие сведения о текущем сетевом сеансе, как количество принятых и отправленных байт, количество принятых cookies и т. д.
- `rundll32.exe localsec.dll, DllRegisterServer` — дает возможность зарегистрировать оснастку Локальные пользователи и группы (`lusrmgr.msc`), позволяющую добавить, удалить или отредактировать группу, к которой принадлежит учетная запись конкретного пользователя.
- `rundll32.exe mycomput.dll, DllRegisterServer` — позволяет зарегистрировать консоль Управление компьютером (`compmgmt.msc`), которая содержит оснастки на все случаи жизни: Просмотр событий, Общие папки, Локальные пользователи и группы, Журналы и оповещение производительности, Диспетчер устройств, Съемные ЗУ и т. д.
- `rundll32.exe SnmpSnap.dll, DllRegisterServer` — выполнение команды приводит к регистрации оснастки расширения SNMP.

ПРИМЕЧАНИЕ

Существует противоположная возможность — удаление сведений о конкретной оснастке. Для этого понадобится воспользоваться командой соответствующей библиотеки DllUnregisterServer.

Если же ни одна из предыдущих команд не помогла, то можно попробовать воспользоваться командой `rundll32.exe MMCNDMGR.DLL, DllRegisterServer`. Она перерегистрирует в реестре саму консоль управления Microsoft, а также некоторые стандартные оснастки данной консоли.

Восстановление Windows

В конце данного раздела рассмотрим несколько команд, которые можно использовать для восстановления многих пользовательских настроек операционной системы Windows к их стандартному состоянию (такими, какими они были в момент установки системы).

- `rundll32.exe SHELL32.dll, DllInstall` — вызов данной команды восстанавливает по умолчанию настройки основных стандартных ActiveX-объектов

системы, а также настройки параметров ветвей реестра `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\StartMenu`, `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\VisualEffects`, `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\Advanced` и т. д.

- `rundll32.exe SYSSETUP.dll, RepairStartMenuItems` — выполнение этой команды восстанавливает по умолчанию содержимое меню Пуск (при этом ссылки на установленные программы не исчезают). По умолчанию устанавливаются параметры ветви системного реестра `HKEY_USERS\.DEFAULT\Software\Microsoft\Windows\CurrentVersion\Explorer\Shell Folders`, определяющей пути к папкам Windows (Мои документы, Мои рисунки, Мои видеозаписи и т. д.). Результат выполнения данной команды можно увидеть на рис. 2.3.

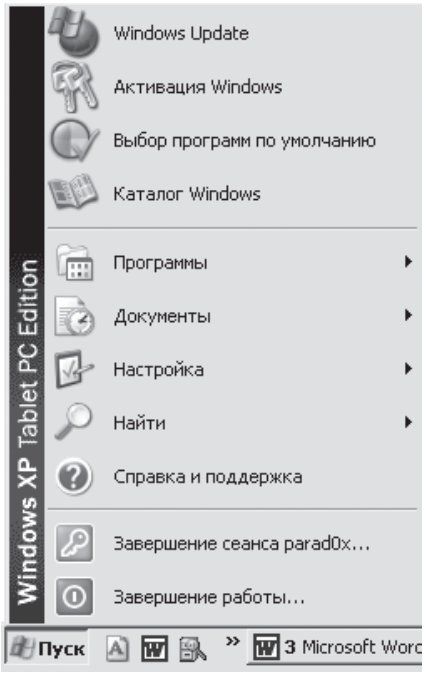


Рис. 2.3. Результат выполнения команды rundll32.exe SYSSETUP.dll, RepairStartMenuItems

- `rundll32.exe SYSSETUP.dll, RunOEMExtraTasks` — вызов данной команды возвращает на Рабочий стол ярлыки Проигрывателя Windows Media и Internet Explorer. Это выполняется путем присваивания значения `yes` параметру строкового типа `DesktopShortcut`, расположенному в ветви системного реестра `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\MediaPlayer\Setup`, а также путем присваивания значения `0` параметру `DWORD`-типа `{871C5380-42A0-1069-A2EA-08002B30309D}`, расположенному в ветвях реестра Windows `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\`

Explorer\HideDesktopIcons\ClassicStartMenu и HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\HideDesktopIcons\ NewStartPanel.

- rundll32.exe SYSSETUP.dll, SetupOobeCleanup — вызов данной команды очищает наиболее важные файлы операционной системы. При этом удаляются и заново создаются (на основе текущих настроек операционной системы) все кусты реестра, а также удаляются различные TMP-файлы.

ВНИМАНИЕ

После применения функций из библиотеки SYSSETUP.dll могут быть заменены стандартные системные файлы, поэтому все установленные в операционной системе заплаты и пакеты обновлений придется переустановить заново.

Файловая система

Теперь рассмотрим несколько команд rundll32.exe, предназначенных для работы с файловой системой Windows. С их помощью можно как создавать файлы, так и удалять уже созданные файлы и каталоги Windows, но нельзя указать содержимое файлов.

Создание файлов

Например, для создания файла можно воспользоваться командой rundll32.exe admparse.dll, CheckDuplicateKeysA «ïóòü è èìÿ ôàéëà ñ ðàñøèðåíèåì». Она имеет один большой недостаток — после запуска вызывает ошибку. Тем не менее со своей работой она справляется — после ее выполнения будет создан или заново переписан указанный в параметре функции файл.

ПРИМЕЧАНИЕ

Например, подобные команды можно использовать для очистки файлов журналов при входе пользователей в систему, указывая их в одном из параметров реестра, предназначенных для автозагрузки.

Удаление файлов и папок

С помощью команд rundll32.exe можно удалить каталог или файл. Для этого достаточно воспользоваться приведенной далее командой: rundll32.exe ADVPACK.dll, DelNodeRunDLL32 «ïóòü ê êàòàëîãó èëè ôàéëó». Функция DelNodeRunDLL32 была написана специально для вызова с помощью команды rundll32.exe, поэтому никаких ошибок при своей работе она не выдает.

Существует еще одна команда, с помощью которой можно выполнить удаление. Но с ее помощью можно удалить только содержимое, а не сам каталог (то есть

переписать необходимый каталог). Это команда `rundll32.exe IEAKENG.dll, BToolbar_SaveA «ïóòü ê ïàïêå»`. Если указанная в данной команде папка уже существует, то она будет автоматически удалена, а потом заново создана.

ПРИМЕЧАНИЕ

Существует один интересный плюс этой команды — с ее помощью можно создать папки даже там, где пользователю это сделать нельзя. Например, в каталоге %userprofile%\Local Settings\Temporary Internet Files\Content.IE5.

Последняя команда, которую мы рассмотрим, — `rundll32.exe WININET.dll, RunOnceUrlCache «ïóòü ê êàòàëîãó»`. Это очень страшная и непредсказуемая команда. Она имеет примерно следующий алгоритм работы: сначала она пытается открыть все содержащиеся в указанной папке файлы и папки. Если при попытке открытия файла или папки система вернула данной команде флаг `FILEATTRIBUTETAGINFORMATION`, то команда удаляет соответствующий файл или папку.

ПРИМЕЧАНИЕ

Данная команда используется браузером Internet Explorer для удаления временных файлов из папки %userprofile%\Local Settings\Temporary Internet Files при закрытии окна браузера, если значение DWORD-параметра Persistent из ветви реестра HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Internet Settings\Cache равно 0.

Выполнение файлов

Теперь рассмотрим несколько команд для выполнения или регистрации файлов. Например, как можно установить INF-файл с помощью команды `rundll32.exe`? Для этого применяется несколько команд, каждая из которых предназначена для отдельной версии INF-файлов.

Чтобы установить обычный INF-файл, необходимо воспользоваться следующей командой — `rundll32.exe setupapi.dll, InstallHinfSection «ðàçäåë â ôàéëå äëÿ íà÷àëà óñòàíîâêè», «ôëàã», «ïóòü è èìÿ ôàéëà»`. При этом «ôëàã» может принимать следующие значения:

- `0` — не перезагружать компьютер после установки INF-файла;
- `1` — перезагружать компьютер после установки INF-файла;
- `2` — спрашивать о перезагрузке компьютера после установки INF-файла;
- `3` — если нужно, то перезагружать компьютер после установки INF-файла;
- `4` — если нужно, то спрашивать о перезагрузке компьютера после установки INF-файла.

Например, с помощью данной команды можно выполнить следующие действия.

- `rundll32.exe setupapi.dll, InstallHinfSection DefaultInstall 132 wsh.inf` — восстановление в реестре настроек сервера сценариев Windows, а также расширений, ему необходимых.
- `rundll32.exe setupapi.dll, InstallHinfSection DefaultInstall 132 sr.inf` — воссоздание в реестре настроек службы Восстановление системы, а также ярлыка программы Восстановление системы в меню Пуск.
- `rundll32.exe setupapi.dll, InstallHinfSection DefaultInstall 132 %17%\PCHealth.inf` — восстановление настроек службы для работы с Центром справки и поддержки.
- `rundll32.exe setupapi.dll, InstallHinfSection DefaultUninstall 132 %17%\PCHealth.inf` — удаление настроек службы для работы с Центром справки и поддержки.
- `rundll32.exe setupapi.dll, InstallHinfSection DefaultInstall 132 %17%\dfrg.inf` — восстановление настроек оснастки `dfrg.msc` и функции `BootDefrag`.
- `rundll32.exe setupapi.dll, InstallHinfSection RestoreBrowserSettings 132 %17%\iereset.inf` — воссоздание настроек браузера Internet Explorer.

Для того чтобы зарегистрировать расширенный INF-файл, вам понадобится команда `rundll32.exe ADVPACK.dll, LaunchINFSectionEx «èìÿ ôàéëà», «èìÿ ðàçäåëà», «èìÿ cab-ôàéëà», «ôëàã»`. Она для своей работы требует как стандартные параметры функции установки (имя INF-файла и раздел в нем, с которого начинается установка), так и специальные параметры.

- Имя CAB-файла, содержащего все файлы, которые устанавливает данный расширенный INF-файл при своей работе.
- Флаг работы процесса установки, наиболее полезные значения которого следующие:
  - `4` — не выводить промежуточные результаты установки INF-файла;
  - `16` — обновить оболочку операционной системы после установки INF-файла;
  - `32` — выполнить резервное копирование данных перед установкой;
  - `64` — выполнить откат установленного INF-файла;
  - `256` — не строить список файлов при установке INF-файла;
  - `512` — при установке INF-файла принудительно задерживать регистрацию ActiveX-объектов, которые должны быть зарегистрированы с помощью данного INF-файла.

Кроме INF-файлов, библиотека `ADVPACK.dll` позволяет также зарегистрировать отдельный ActiveX-объект. Каждый ActiveX-объект поставляется в виде файла

с расширением OCX, регистрация которого происходит с помощью такой команды: `rundll32.exe ADVPACK.dll, RegisterOCX «ïóòü è èìÿ ôàéë .ocx»`.

Теперь рассмотрим набор команд `rundll32.exe`, предназначенных для открытия файлов Windows, имеющих специальные расширения и содержимое.

- `rundll32.exe CdfView.dll, OpenChannel «ïóòü è èìÿ ôàéëà êàíàëà»` — открыть данный файл канала.
- `rundll32.exe CdfView.dll, Subscribe «ïóòü è èìÿ ôàéëà ïîäïèñ÷èêà»` — сделать данный канал доступным автономно.
- `rundll32.exe CRYPTEXT.dll, CryptExtAddCER «ôàéë ñåðòèôèêàòà áåçîïàñíîñòè»` — добавить данный файл сертификата безопасности.
- `rundll32.exe CRYPTEXT.dll, CryptExtAddCRL «ôàéë ñïèñêà îòçûâîâ ñåðòèôèêàòîâ»` — присоединить данный список отзыва сертификатов.
- `rundll32.exe CRYPTEXT.dll, CryptExtAddCTL «ôàéë ñïèñêà äîâåðèÿ ñåðòèôèêàòîâ»` — добавить данный список доверия сертификатов.
- `rundll32.exe CRYPTEXT.dll, CryptExtAddP7R «ôàéë îòâåòà íà çàïðîñ ñåðòèôèêàòà»` — присоединить данный файл ответа на запрос сертификата.
- `rundll32.exe CRYPTEXT.dll, CryptExtAddPFX «ôàéë îáìåíà ëè÷íîé èíôîðìàöèåé»` — добавить данный файл обмена личной информацией.
- `rundll32.exe CRYPTEXT.dll, CryptExtAddSPC «ôàéë ñåðòèôèêàòà PCKS #7»` — присоединить данный файл сертификата PCKS #7.
- `rundll32.exe CRYPTEXT.dll, CryptExtOpenCAT «ôàéë êàòàëîãà áåçîïàñíîñòè»` — открыть данный файл каталога безопасности.
- `rundll32.exe CRYPTEXT.dll, CryptExtOpenCER «ôàéë ñåðòèôèêàòà áåçîïàñíîñòè»` — открыть указанный файл сертификата безопасности.
- `rundll32.exe CRYPTEXT.dll, CryptExtOpenCRL «ôàéë ñïèñêà îòçûâîâ ñåðòèôèêàòîâ»` — открыть данный файл списка отзывов сертификатов.
- `rundll32.exe CRYPTEXT.dll, CryptExtOpenCTL «ôàéë ñïèñêà äîâåðèÿ ñåðòèôèêàòîâ»` — открыть указанный файл списка доверия сертификатов.
- `rundll32.exe CRYPTEXT.dll, CryptExtOpenP10 «ôàéë çàïðîñà íà ñåðòèôèêàò»` — открыть данный файл запроса на сертификат.
- `rundll32.exe CRYPTEXT.dll, CryptExtOpenP7R «ôàéë îòâåòà íà çàïðîñ ñåðòèôèêàòà»` — открыть указанный файл ответа на запрос сертификата.
- `rundll32.exe CRYPTEXT.dll, CryptExtOpenPKCS7 «ôàéë ñåðòèôèêàòà PCKS #7»` — открыть данный файл сертификата PCKS #7.
- `rundll32.exe CRYPTEXT.dll, CryptExtOpenSTR «ôàéë õðàíèëèùà ñåðòèôèêàòîâ»` — открыть указанный файл хранилища сертификатов.

- rundll32.exe dsquery.dll, OpenSavedDsQuery «ôàéë çàïðîñà ê ñëóæáå êàòàëîãîâ ActiveDirectory» — при вызове команды происходит попытка выполнить файл запроса к каталогу Active Directory.
- rundll32.exe msconf.dll, NewMediaPhone «ïóòü ê ôàéëó» — открыть данный файл телефонии. Эта команда используется программой NetMeeting для открытия соответствующих файлов.
- rundll32.exe msconf.dll, OpenConfLink «ïóòü ê ôàéëó» — команда также используется программой NetMeeting. Именно с помощью этой команды создаются ярлыки различных конференций (для автоматического соединения с другими сетевыми компьютерами).
- rundll32.exe netshell.dll, InvokeDunFile — открыть DUN-файл (Dialup Networking File).
- rundll32.exe SHDOCVW.dll, OpenURL «ïóòü ê ôàéëó» — открыть файл ярлыка Интернета (имеет расширение URL), указанный в качестве параметра функции.
- rundll32.exe shell32.dll, Control_RunDLL «èìÿ CPL èëè DLL-ôàéëà» — вызов данной команды приводит к запуску соответствующего CPL-файла или определенной функции файла DLL. Например, после вызова команды rundll32.exe shell32.dll, Control_RunDLL main.cpl откроется апплет Мышь. Если вы разочарованы, то могу сказать еще об одной особенности работы данной функции — с ее помощью можно указать вкладку, на которой будет открыт апплет. Например, вызов команды rundll32.exe shell32.dll, Control_RunDLL main.cpl,,2 приведет к открытию апплета Мышь на вкладке Параметры указателя. Аналогично можно открывать и любые другие вкладки различных апплетов (при этом вкладки нумеруются, начиная с нуля). Функция Control_RunDLL используется и в таких командах:
  - rundll32.exe shell32.dll, Control_RunDLL desk.cpl desk, @Appearance — открыть диалоговое окно Свойства: Экран на вкладке Оформление;
  - rundll32.exe shell32.dll, Control_RunDLL desk.cpl desk, @Appearance /Action:OpenMSTheme /file:«ïóòü ê ôàéëó òåìû ðàáî÷åãî ñòîëà» и rundll32.exe shell32.dll, Control_RunDLL desk.cpl desk, @Appearance /Action:OpenTheme /file:«ïóòü ê ôàéëó òåìû ðàáî÷åãî ñòîëà» — установить файл стиля оформления Windows XP;
  - rundll32.exe shell32.dll, Control_RunDLL desk.cpl desk, @Desktop — открыть диалог Свойства: Экран на вкладке Рабочий стол;
  - rundll32.exe shell32.dll, Control_RunDLL desk.cpl desk, @Settings — открыть диалог Свойства: Экран на вкладке Параметры;
  - rundll32 shell32.dll, Control_RunDLL NetSetup.cpl,@0, WNSW — отобразить окно Мастера беспроводной сети;
  - rundll32 shell32.dll, Control_RunDLL NetSetup.cpl — отобразить окно Мастера настройки сети.

Особо стоит сказать о двух командах `Control_RunDLL`, использующих для своей работы библиотеки Windows:

- `rundll32.exe shell32.dll, Control_RunDLL hotplug.dll` — вызов данной команды отображает диалог «горячего» удаления внешнего устройства (рис. 2.4).

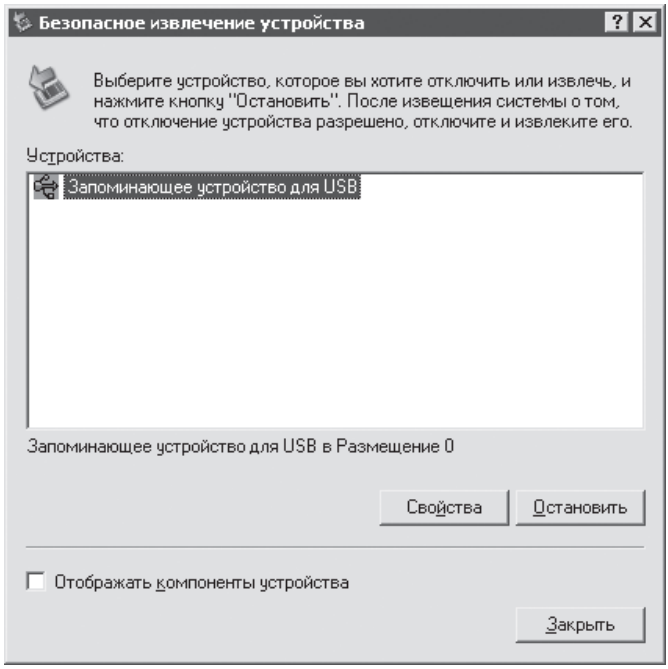


Рис. 2.4. Результат выполнения команды rundll32.exe shell32.dll, Control_RunDLL hotplug.dll

- `rundll32.exe shell32.dll, Control_RunDLL input.dll` — позволяет вызвать диалог Язык и службы текстового ввода.

ПРИМЕЧАНИЕ

При выполнении приведенной выше команды без указания какого-либо CPL или DLL-файла будет открыто окно Панель управления.

- `rundll32.exe shell32.dll, OpenAs_RunDLL «ïóòü è èìÿ ôàéëà»` — с помощью данной команды можно отобразить диалоговое окно Открыть с помощью для открытия указанного в параметре функции файла. Например, можно открыть исполняемый файл программы в Блокноте или с помощью другой программы (по умолчанию исполняемые файлы нельзя вызывать с помощью диалога Открыть с помощью).

- rundll32.exe shimgvw.DLL, ImageView_Fullscreen «ïóòü ê ôàéëó èçîáðàæåíèÿ» — позволяет открыть указанный файл изображения с помощью программы просмотра изображений и факсов.

Выполнение команд

Существует еще одна интересная возможность, которую можно использовать при разработке файлов сценариев, — выполнение команд, записанных в ветви реестра. Для этого применяются функции библиотеки ADVPACK.dll. Например, после выполнения команды rundll32.exe ADVPACK.dll, UserInstStubWrapper «ïîäðàçäåë» система выполнит строку, содержащуюся в параметре строкового типа RealStubPath, расположенном в ветви реестра HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Active Setup\Installed Components\«ïîäðàçäåë» (по умолчанию параметр отсутствует). Одновременно плюсом и минусом данной команды является то, что программа, которую вы запустите после обработки содержимого параметра RealStubPath, будет запущена как процесс, то есть ей будет отказано во взаимодействии с Рабочим столом и она не сможет отобразить свое окно. Минус этого ясен, а плюс можно определить на примере. Если указать в параметре RealStubPath ветви системного реестра HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Active Setup\Installed Components\para следующую команду: rundll32.exe amovie.ocx, RunDll /play /close e:\music\B-2\âîëêè.wma, которая уже была описана в гл. 1, то после вызова команды rundll32.exe ADVPACK.dll, UserInstStubWrapper para будет воспроизведен данный музыкальный файл. При этом проигрывание данного файла можно будет прекратить только выгрузкой из памяти процесса rundll32.exe, который породил это воспроизведение, ведь никакого окна индикации открытого файла отображено не будет.

Можно также воспользоваться разновидностью приведенной выше команды — rundll32.exe ADVPACK.dll, UserUnInstStubWrapper «ïîâêàòàëîã». После ее вызова будет выполнена строка параметра RealStubPath, расположенного в ветви реестра HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Active Setup\Installed Components\«ïîâêàòàëîã».Restore. Например, выполнение команды rundll32.exe ADVPACK.dll, UserUnInstStubWrapper para будет использовать параметр из ветви реестра HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Active Setup\Installed Components\para.Restore.

Кроме функций библиотеки ADVPACK.dll, существует еще одна возможность запуска команд из реестра — использование ветви реестра HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\RunOnceEx (данная ветвь присутствует и в корневом разделе HKEY_LOCAL_MACHINE). По умолчанию ее содержимое выполняется только при следующем входе пользователя в систему, хотя существует возможность сказать системе, чтобы она обработала содержимое этой ветви немедленно. При этом все команды из ветви после своего выполнения автоматически удаляются. Следует также учитывать одну особенность ветви — если

в ней будет определено несколько запусков программ, то все они будут запускаться последовательно. Сначала запустится первая программа, после того, как пользователь ее закроет, запустится вторая программа и т. д.

Формат содержимого данной ветви немного отличается от формата подобных ей ветвей, направленных на выполнение команд при входе пользователя в систему (например, ветви реестра `HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Run`). Каждая программа или команда, которая должна быть запущена, записывается в отдельный раздел данной ветви реестра — название раздела не имеет значения, но лучше последовательно пронумеровать эти разделы, чтобы система могла легко определить, какая программа должна запускаться первой. Параметр `Ïî óìîë÷àíèþ` каждого раздела определяет название, которое будет отображаться в диалоге индикации установки при выполнении программы, определенной в соответствующем разделе реестра. При этом если параметр `Ïî óìîë÷àíèþ` не определен, то в диалоге индикации не будет никакой информации о запуске данной программы или команды, хотя она будет запущена. Сама же программа или команда записывается в значении строкового параметра соответствующего раздела (название параметра не имеет значения). В разделе можно определить несколько команд, записав их в несколько строковых параметров, — все эти команды будут выполняться последовательно на данном шаге установки.

Чтобы лучше понять формат данной ветви реестра, рассмотрим один пример. Ниже в листинге приведен пример записей ветви реестра `HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\ RunOnceEx`, экспортированных в REG-файл.

Листинг 2.1. Содержимое ветви реестра RunOnceEx

```
Windows Registry Editor Version 5.00

[HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\RunOnceEx\1]
@="Запуск cmd.exe и calc.exe"
"cmd"="cmd.exe"
"calc"="calc.exe"

[HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\RunOnceEx\2]
@="Запуск блокнота"
"notepad.exe"="notepad.exe"
```

Допустим, содержимое приведенного листинга находится в реестре. Теперь нужно ввести команду `rundll32.exe IERNONCE.dll, RunOnceExProcess`. Она указывает системе, что та должна обработать содержимое ветви системного реестра `HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\RunOnceEx`. Результат обработки данной ветви можно увидеть на рис. 2.5.

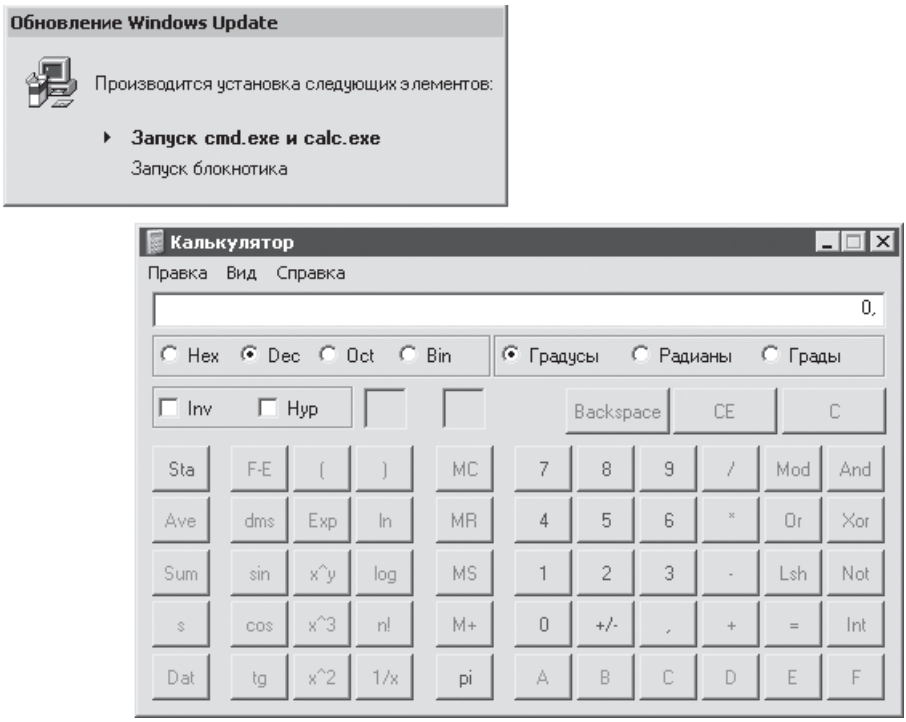


Рис. 2.5. Способ работы запуска команд из ветви реестра HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\RunOnceEx

Как видно на рисунке, при запуске команды `rundll32.exe` появляется диалог индикации, в котором указано, какой шаг установки сейчас выполняется. При этом был запущен Калькулятор, так как при сортировке между названиями параметров в разделе `1` Калькулятор, судя по названию параметра, запускается первым. После того как вы закроете окно Калькулятора, будет открыто окно командного процессора. При этом индикатор будет все еще указывать на первый шаг, ведь еще не началась обработка содержимого следующего раздела ветви (раздела `2`). После того как вы закроете окно командного процессора, будет запущен Блокнот, и при этом индикатор начнет указывать на второй шаг операции установки.

Если вы не укажете значений параметров по умолчанию ни в одном из разделов данной ветви реестра, то индикатор установки отображаться не будет. Тем не менее сами команды будут выполняться.

Другие операции

В данном разделе будут перечислены некоторые команды `rundll32.exe`, которые не вошли ни в один из предыдущих разделов, но тем не менее с их помощью можно выполнить те или иные действия.

- `rundll32.exe INITPKI.dll, InitializePKI` — позволяет установить поддержку файлов расширений сертификатов и отношения доверия в Active Directory.
- `rundll32.exe mobsync.dll, RegSetUserDefaults` — с помощью данной команды можно быстро восстановить стандартные настройки синхронизации текущего пользователя при помощи сетевого соединения LAN. Все эти настройки находятся в ветви реестра `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Syncmgr\AutoSync\«èìÿ ïîëüçîâàòåëÿ»`, которую команда и переписывает при своем запуске.
- `rundll32.exe mshtml, PrintHTML «ïóòü ê ôàéëó»` — с помощью этой команды можно вызвать диалог Печать для печати указанного файла.
- `rundll32 printui.dll, PrintUIEntry /s` — позволяет отобразить диалоговое окно Свойства: Сервер печати. Аналогичное окно можно вызвать и с помощью контекстного меню папки Принтеры и факсы. Для этого в данном контекстном меню нужно выбрать команду Свойства сервера.
- `rundll32.exe MSI39.dll, VMAskDisableAutorun` — очень интересная команда, которая используется при установке программы VMware. Сейчас практически повсюду можно услышать совет о том, как с помощью реестра отключить функцию автозапуска компакт-дисков — для этого нужно `DWORD`-параметру `AutoRun`, расположенному в ветви реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Cdrom`, присвоить значение `0`. Можно это сделать и без обращения к реестру — достаточно вызвать данную команду, после чего перед вами отобразится диалог с вопросом, действительно ли вы хотите отключить автозапуск дисков (рис. 2.6), на который нужно ответить Yes. Следует только учитывать, что если автозапуск уже отключен, то никакого диалога выведено не будет.

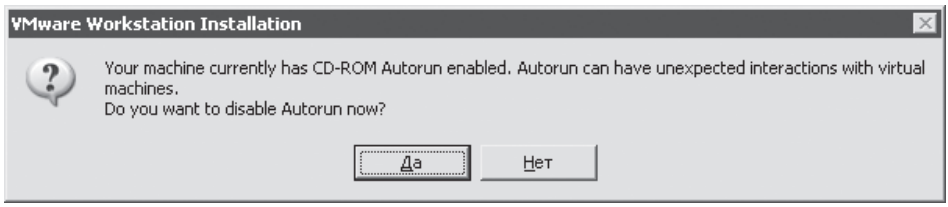


Рис. 2.6. Результат выполнения команды rundll32.exe MSI39.dll, VMAskDisableAutorun

Если же вы уже запретили автозапуск компакт-дисков, то с помощью еще одной команды — `rundll32.exe MSI39.dll, VMRestoreRegistry` — можно включить его снова.

- `rundll32.exe NETPLWIZ.dll, ClearAutoLogon` — еще одна интересная команда. Сейчас стало модно конфигурировать на компьютере автоматический вход в систему с правами указанного пользователя. Как вызвать диалог для этого, было рассказано ранее, сейчас же мы решим задачу отключения данной функции. Функция полезна лишь в домашних условиях — в корпоративной сети ее установка на клиентских компьютерах может создать большую брешь в среде

безопасности предприятия. Если вам все время приходится проверять компьютеры на автоматический вход в систему, то для облегчения своей работы вы можете воспользоваться данной командой, ведь после ее вызова функция автоматического входа в систему будет отключена.

- `rundll32.exe NETPLWIZ.dll, SHDisconnectNetDrives` — вызов данной команды приводит к отключению всех сетевых дисков (дисков, которые создаются с помощью команды Подключить сетевой диск в меню Сервис папки или в контекстном меню значка Мой компьютер). Если на данный момент ни один диск не подключен, то будет выведено сообщение о том, что сетевых дисков, которые можно отключить, нет.

ПРИМЕЧАНИЕ

Сам же мастер подключения сетевых дисков можно вывести не только с помощью меню Сервис или контекстного меню значка Мой компьютер, но и с помощью команды rundll32.exe shell32.dll, SHHelpShortcuts_RunDLL Connect. При этом с помощью команды rundll32.exe shell32.dll, SHHelpShortcuts_RunDLL Disconnect можно выполнить отключение дисков.

- `rundll32.exe netshell.dll, DoInitialCleanup` — выполнение данной команды удаляет содержимое ветви реестра `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\NetworkCards`.
- `rundll32.exe NTPRINT.dll, PSetupKillBadUserConnections` — вызов данной команды очищает неработающие соединения с принтерами. Список всех соединений с принтерами можно найти в ветви реестра `HKEY_CURRENT_USER\Printers\Connections`, которую как раз и просматривает данная команда. При этом список неработающих соединений находится в параметре строкового типа `Bad Connections`, расположенном в ветви реестра `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Print`.
- `rundll32.exe NWCFG.dll, CleanupRegistryForNWCS` — позволяет очистить реестр от настроек NWC (оболочка расширения для NetWare). После выполнения команды параметру `NwcsInstalled` в ветви `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\NWCS` присваивается значение `0` (имеет тип `DWORD`). При запуске данной команды удаляется содержимое ветвей реестра `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Shell Extensions\Approved` (только несколько параметров) и `HKEY_CLASSES_ROOT\Network\Type\3`, а также сведения об ActiveX-объектах.

  Если же вам необходимо выполнить обратную операцию, то есть установить или перерегистрировать в реестре информацию об NWC, то можно воспользоваться командой `rundll32.exe NWCFG.dll, SetupRegistryForNWCS`, которая именно для этого и предназначена.
- `rundll32.exe NWPROVAU.dll, NwCleanupGatewayShares` — выполнение этой команды очищает список общих папок для клиентов NetWare. Список расположен в двух ветвях реестра — `HKEY_LOCAL_MACHINE\SYSTEM\`

CurrentControlSet\Services\NWCWorkstation\Shares и HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\NWCWorkstation\Drives. Содержимое именно этих ветвей и удаляется соответствующей командой.

- rundll32.exe powrprof.dll, SetSuspendState — вызов данной команды переводит компьютер в состояние спящего режима (если этот режим поддерживается оборудованием и включен на вкладке Спящий режим диалога Электропитание). Спящий режим — это режим работы компьютера, при котором все содержимое памяти сбрасывается на жесткий диск и компьютер выключается. Плюсом является быстрый переход между спящим режимом и обычным состоянием компьютера — в зависимости от объема оперативной памяти, установленной на компьютере, после подачи питания до входа в систему проходит от нескольких секунд до нескольких десятков секунд (не учитывая время на самотестирование компьютера). При этом пропускаются такие шаги загрузки, как выбор операционной системы (если на компьютере используется несколько операционных систем) и собственно регистрация пользователя в системе — просто содержимое сброшенной на жесткий диск памяти помещается обратно в оперативную память, и работа системы продолжается, как будто никакого спящего режима не было.
- rundll32.exe RASAPI32.dll, RasSetSharedAutoDial — выполнение данной команды присваивает DWORD-параметру SharedAutoDial значение, равное 1. Он находится в ветви реестра HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\SharedAccess\Parameters.
- rundll32.exe rasdlg.dll, RasUserEnableManualDial — выполнение данной команды присваивает DWORD-параметру OperatorDial значение, равное 1. Он находится в ветви реестра HKEY_USERS\.DEFAULT\Software\Microsoft\RAS Logon Phonebook.
- rundll32.exe rasman.dll, RasDoIke — выполнение данной команды может привести к частичному зависанию компьютера. Сначала нельзя получить доступ к Панели задач и меню Пуск, а также ко всем открытым приложениям. При попытке открытия Диспетчера задач происходит полное зависание компьютера. При этом такие программы, как Проигрыватель Windows Media, будут работать, будет возможно обращаться к общедоступным папкам компьютера, но нельзя будет получить доступ к оболочке операционной системы.
- rundll32.exe shell32.dll, Control_FillCache_RunDLL — позволяет выполнить начальную инициализацию Панели управления.
- rundll32.exe SPOOLSS.DLL, UpdatePrinterRegAll — с помощью этой команды можно обновить в реестре все сведения о принтерах системы для всех пользователей. Команда удаляет (и заново создает на основе текущих данных) следующие ветви реестра: HKEY_CURRENT_USER\Printers\DevModePerUser, HKEY_CURRENT_USER\Software\Microsoft\Windows NT\CurrentVersion\Devices, HKEY_CURRENT_USER\Software\Microsoft\Windows NT\CurrentVersion\PrinterPorts.
- rundll32.exe sti_ci.dll, ?CreateWiaShortcut@@YGHXZ — выполнение команды приводит к созданию в списке Стандартные меню Пуск ярлыка Мас-

тера работы со сканером или цифровой камерой. Существует также возможность автоматического удаления созданного ярлыка. Для этого используется команда `rundll32.exe sti_ci.dll, ?DeleteWiaShortcut@@YGHXZ`.

ПРИМЕЧАНИЕ

Ярлык Мастера работы со сканером или цифровой камерой можно создать и с помощью команды rundll32.exe sti_ci.dll, WiaCreateWizardMenu.

- `rundll32.exe syncui.dll, Briefcase_Create` — с помощью этой команды на Рабочем столе можно создать папку Портфель, которая по своей функциональности напоминает автономные файлы. При подключении ноутбука или карманного компьютера (КПК) при использовании папки Портфель система будет сверять ее содержимое на ноутбуке с содержимым на настольном компьютере. Если в данной папке на компьютере (или ноутбуке) будут найдены новые версии файлов, они автоматически заменят собой более старые версии, хранящиеся в папке, расположенной на ноутбуке (или на компьютере соответственно).
- `rundll32.exe url.dll, FileProtocolHandler «ïóòü ê êàòàëîãó»` — вызов данной команды приводит к открытию в Проводнике соответствующей папки. Возможен вызов данной команды без параметра — в этом случае будет открыта папка профиля текущего пользователя (`%userprofile%`).
- `rundll32.exe url.dll, TelnetProtocolHandler «IP-àäðåñ óäàëåííîãî êîìïüþòåðà»` — позволяет подключиться с помощью telnet к указанному вами удаленному компьютеру.
- `rundll32.exe user32.dll, LockWorkStation` — с помощью этой команды можно заблокировать работу компьютера. Теперь доступ к компьютеру можно будет получить, только если ввести пароль текущего зарегистрированного в системе пользователя или пароль учетной записи администратора компьютера.
- `rundll32.exe USER32.dll, mouse_event` — необычная команда. Первый ее запуск эквивалентен нажатию (но не отпусканию) правой кнопки мыши. После выполнения этой команды в первый раз при перемещении указателя мыши будет отображаться прямоугольник выделения. Второй же запуск данной команды эквивалентен нажатию правой кнопки мыши, то есть отображается контекстное меню.

ПРИМЕЧАНИЕ

Эта команда не всегда работает корректно. Иногда сразу при первом запуске может отобразиться контекстное меню, а иногда вообще ничего не происходит.

- `rundll32.exe user32.dll, SetCursorPos` — вызов данной команды приводит к установке указателя в правый нижний угол экрана.

- `rundll32.exe user32.dll, SwapMouseButton` — с помощью данной команды можно поменять местами функциональность левой и правой кнопок мыши, то есть левая кнопка мыши будет открывать контекстное меню, а правая — выделять значки. Обратно сменить функциональность клавиш мыши с помощью данной команды нельзя. Это же можно сделать с помощью апплета Мышь — достаточно установить или снять флажок Обменять назначение кнопок на вкладке Кнопки мыши.
- `rundll32.exe w32time.dll, W32TimeVerifyJoinConfig` — дает возможность присвоить параметрам DWORD-типа `MaxNegPhaseCorrection`, а также `MaxPosPhaseCorrection`, расположенным в ветви системного реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W32Time\Config`, значения `0xffffffff`.

  Этим параметрам можно также присвоить значения `0xD2F0`. Для этого используется команда `rundll32.exe w32time.dll, W32TimeVerifyUnjoinConfig`.

Глава 3

Программы

- **Internet Explorer**
- **Outlook Express**
- **Другие программы**

В этой главе будут рассмотрены некоторые интересные команды, используемые различными стандартными приложениями, поставляемыми вместе с операционной системой Windows. К таким приложениям в первую очередь можно отнести Internet Explorer и Outlook Express.

Internet Explorer

Internet Explorer — браузер, входящий в стандартную поставку любой версии операционной системы Windows XP. Более того, он является ее неотъемлемой частью, и, как бы данный браузер ни критиковали, большая часть пользователей Интернета для доступа к Сети использует именно его. В поставку Internet Explorer входит браузер версии 6.0, поэтому команды `rundll32.exe`, которые будут описаны в данном разделе книги, содержатся в библиотеках именно этой версии браузера. Хотя это не значит, что более ранние версии Internet Explorer не будут поддерживать этих команд.

Оболочка

Теперь рассмотрим команды `rundll32.exe`, предназначенные для взаимодействия с пользователем. Раньше при описании файла `inetcpl.cpl` уже упоминались такие команды, но, кроме них, существует также некоторое количество команд, предоставляемых стандартными библиотеками Windows.

Первой из этих команд является `rundll32.exe IEAKENG.dll, ModifyZones`. С ее помощью можно отобразить диалог Свойства обозревателя, в котором будут доступны только две вкладки — Безопасность и Конфиденциальность. Диалог не является какой-то новой разновидностью окна Свойства обозревателя, отображаемого после выбора команды Свойства обозревателя меню Сервис, — это все тот же диалог, доступ к остальным вкладкам которого был запрещен при запуске команды с помощью групповых политик. После нажатия кнопки ОК данного окна настройки групповых политик опять примут свой стандартный вид — все изменения, которые были сделаны командой, будут удалены.

ПРИМЕЧАНИЕ

Если данная команда будет использоваться учетной записью, которой запрещено изменять групповые политики, то вызываемый диалог будет содержать все вкладки.

Если вас удивила предыдущая команда, то вы еще больше удивитесь, когда узнаете о команде `rundll32.exe IEAKENG.dll, ShowInetcpl`. При ее выполнении операционная система с помощью групповых политик сначала запрещает доступ ко всем вкладкам диалога Свойства обозревателя, а потом пытается открыть этот диалог. При этом, как и следовало ожидать, у команды ничего не получается

и она выдает сообщение о том, что данный диалог запрещен администратором. После этого команда удаляет все изменения групповых политик, которые она выполняла в начале своей работы.

ПРИМЕЧАНИЕ

Если данная команда, как и предыдущая, будет применяться пользователем, для которого запрещено изменение групповых политик, то диалог Свойства обозревателя будет открыт.

Конфигурация

И наконец-то команды `rundll32.exe` для настройки конфигурации Internet Explorer. Этих команд не очень много, но они могут быть полезны в некоторых случаях. Например, если вы изменили настройки брэндов Internet Explorer (логотип в правом верхнем углу браузера, высота и фон панели инструментов) с помощью параметров реестра, описанных в части 2, посвященной работе с реестром Windows, и теперь хотели бы восстановить стандартные настройки, то нет необходимости пользоваться реестром. Достаточно выполнить команду `rundll32.exe iedkcs32.dll, BrandCleanInstallStubs`, после чего браузер пересоздаст ветви реестра, описывающие используемые им брэнды. Можно также воспользоваться командой `rundll32.exe iedkcs32.dll, Clear`. Но в этом случае будут переписаны изменения не только брэндов, но и других параметров браузера Internet Explorer.

Еще одной проблемой, которую позволяют решить команды `rundll32.exe`, является возможность перезаписи стандартных пунктов меню Избранное браузера Internet Explorer (ссылки на MSN.com, Windows Media, Hotmail, Программы радиопередач). Если вы удалили эти ссылки и теперь вам необходимо получить доступ к одному из перечисленных сайтов, то можно просто воспользоваться командой `rundll32.exe iedkcs32.dll, BrandIE4 SIGNUP`. После этого стандартные ссылки меню Избранное будут созданы заново.

Вот, собственно, и все. Но в конце данного раздела рассмотрим некоторые дополнительные команды `rundll32.exe`, которые вам могут понадобиться.

- `rundll32.exe SHDOCVW.dll, SetShellOfflineState` — вызов данной команды приводит к установке флажка Работать автономно в меню Файл, после чего при следующем запуске браузера Internet Explorer попытка подключения к Интернету осуществляться не будет — она произойдет только после снятия этого флажка или после ввода адреса сайта в адресной строке.
- `rundll32.exe WININET.dll, DeleteIE3Cache` — позволяет удалить содержимое каталогов `cache1`, `cache2`, `cache3` и `cache4`, расположенных по адресу `%userprofile%\Local Settings\Temporary Internet Files\Content.IE5`.

- `rundll32.exe WININET.dll, InternetClearAllPerSiteCookieDecisions` — вызов данной команды приводит к очищению содержимого ветви `HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Internet Settings\P3P\History`.
- `rundll32 INETCFG.dll InetSetAutoProxyA «IP-àäðåñ»` — с помощью данной команды можно установить флажок Использовать сценарий автоматической настройки в диалоговом окне Настройка локальной сети, а также записать в поле Адрес данного диалогового окна значение указанного в вызываемой команде IP-адреса. Диалоговое окно Настройка локальной сети вызывается нажатием кнопки Настройка LAN на вкладке Подключения диалога Свойства обозревателя.

  В контексте системного реестра Windows изменяются значения параметров `AutoProxyDetectMode` (параметр `REG_BINARY`-типа становится равен `1`) и `AutoConfigURL` (параметр строкового типа, хранящий значение указанного IP-адреса). Оба этих параметра находятся в ветви реестра `HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Internet Settings`.

Настройки ограничений

Отдельно стоит сказать о возможностях работы с ограничениями, которые можно настроить для закрытия доступа к страницам, содержащим насилие или другие запретные темы. Доступ к данным настройкам можно получить после нажатия кнопки Настройка на вкладке Содержание диалога Свойства обозревателя. После нажатия этой кнопки перед вами появится диалог Ограничение доступа (или будет выведен запрос на ввод пароля, если возможность ограничения уже включена), открытый на вкладке Оценки, с помощью которой можно настроить уровень ограничения доступа (перемещая ползунок на уровень, который соответствует вашим представлениям о морали). На вкладке Разрешенные узлы можно указать страницы, на которые не будут распространяться ограничения доступа. На вкладке Общие определяется пароль для доступа к ограничениям, а также файл системы оценок, который будет определять дополнительные параметры того, к какому сайту можно подключиться, а к какому нет (по умолчанию в Windows существует только один файл оценок, но новые можно создать или скачать в Интернете). На вкладке Дополнительно определяется сайт, который будет использоваться для получения инструкций системой оценки (если ей необходимы инструкции), а также дополнительный файл правил для доступа к сайтам, являющийся последним рубежом между вами и нежелательными сайтами.

Теперь посмотрим, какие возможности предоставляют команды `rundll32.exe` при работе с ограничением доступа.

В первую очередь с помощью команд `rundll32.exe` можно открыть диалог Ограничение доступа. Для этого используется команда `rundll32.exe IEAKENG.dll, ModifyRatings`, результат выполнения которой приведен на рис. 3.1. Аналогичных действий можно добиться, если воспользоваться командой `rundll32.exe MSRATING.dll, RatingSetupUI`.

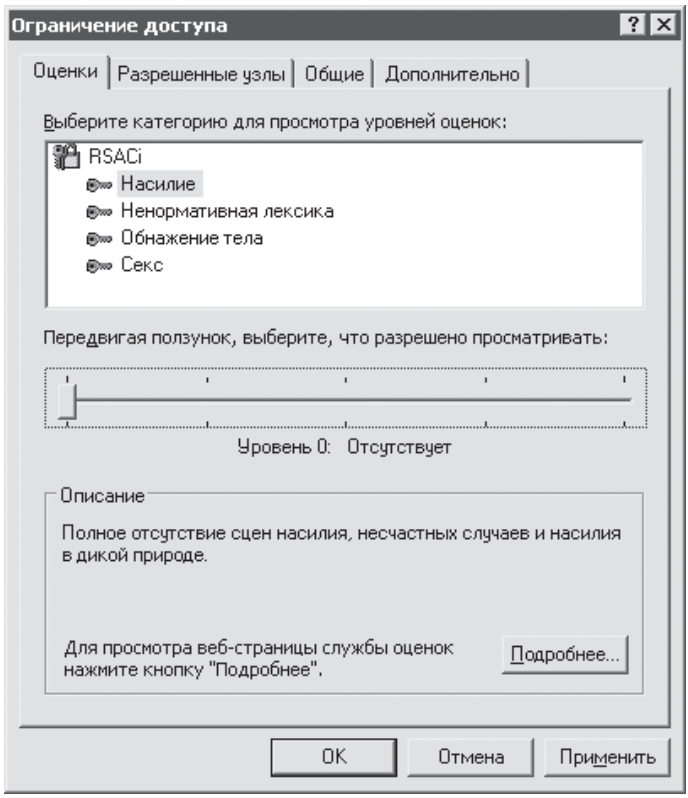


Рис. 3.1. Результат выполнения команды rundll32.exe IEAKENG.dll, ModifyRatings

Существует возможность быстрого открытия диалога Системы оценок (рис. 3.2). Для этого используется команда `rundll32.exe MSRATING.dll, ClickedOnRAT`, после выполнения которой будет открыт диалог Ограничение доступа на вкладке Общие, а также Системы оценок.

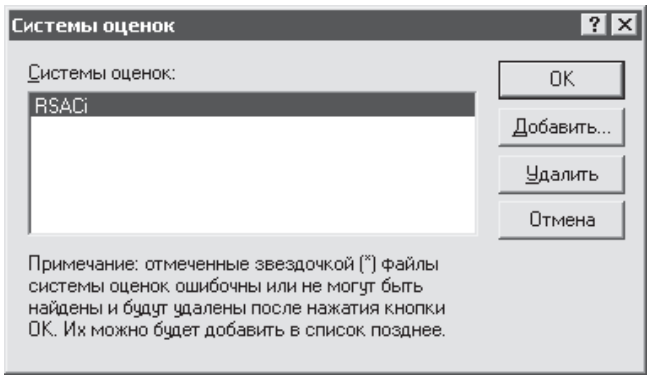


Рис. 3.2. Результат выполнения команды rundll32.exe MSRATING.dll, ClickedOnRAT

Существует еще одна интересная возможность — установка файла правил PICS с помощью команды `rundll32.exe MSRATING.dll, ClickedOnPRF «ïóòü ê ôàéëó ïðàâèë PICS»`. После вызова данной команды произойдет открытие диалога Ограничение доступа (на вкладке Дополнительно) и браузер попытается импортировать указанный вами файл правил PICS в реестр.

Наконец, если вы уже настроили ограничения доступа, но теперь необходимо установить такие же ограничения на другом компьютере, то можно воспользоваться командой `rundll32.exe IEAKENG.dll, ImportRatingsA «ïóòü è èìÿ inf-ôàéëà»`. После ее выполнения (не всегда выполняется с первого раза) будет создан INF-файл, включающий большую часть настроек ограничений доступа (разрешенные и запрещенные узлы, пароль, подсказка, правила PICS и т. д.). В указанной папке будет создан INF-файл с именем `ratrsop.inf`, содержащий дополнительные настройки.

После применения созданных файлов вы можете воспользоваться следующей командой: `rundll32.exe MSRATING.dll, RatingEnable` — для быстрого включения созданных вами ограничений доступа. После этого операционная система попросит вас ввести пароль и при правильном пароле включит ограничения.

ПРИМЕЧАНИЕ

Существует возможность перерегистрации параметров реестра, используемых для определения расширений файлов для файлов правил PICS и систем оценок. Для этого достаточно воспользоваться командой rundll32.exe MSRATING.dll, DllRegisterServer. Можно также применить команду rundll32.exe MSRATING.dll, DllUnregisterServer. В этом случае все сведения о расширениях файлов, используемых функцией ограничения доступа, будут удалены из реестра.

Outlook Express

Как и Internet Explorer, Outlook Express входит в стандартную поставку операционной системы Windows и является почтовым клиентом, предназначенным для отправки и получения писем с почтового сервера, такого как, например, www.mail.ru. Outlook Express имеет намного меньше параметров `rundll32.exe`, которые могут быть интересны, но тем не менее о них стоит рассказать.

ПРИМЕЧАНИЕ

Некоторые команды для своей работы используют библиотеки, расположенные в том же каталоге, что и файл msimn.exe. В этом случае в описываемой команде rundll32.exe будет применяться стандартный путь к каталогу почтового клиента — %programfiles%\Outlook Express. Если в вашей системе используется другой путь, то при определении команды нужно будет указать его.

Открытие почтового клиента

Почтовый клиент Outlook Express также можно запустить с помощью команды `rundll32.exe`. Для этого применяется следующая команда: `rundll32.exe "%programfiles%\Outlook Express\MSOE.DLL", CoStartOutlookExpress`. Она просто открывает окно Outlook Express, но имеет один большой недостаток — довольно часты случаи, когда окно после своего открытия автоматически закрывается (если установлена возможность автоматического завершения Outlook Express при возникновении ошибки).

Более интересной возможностью, которую предоставляет почтовый клиент, является реализуемая командой `rundll32.exe "%programfiles%\Outlook Express\MSOE.DLL", MAPISendDocuments` возможность открытия диалогового окна Вложенные файлы, отображенного на рис. 3.3. С его помощью можно создать и отправить письмо, при этом темой письма будет Вложенные файлы (хотя никто вам не помешает изменить тему, если это будет нужно).

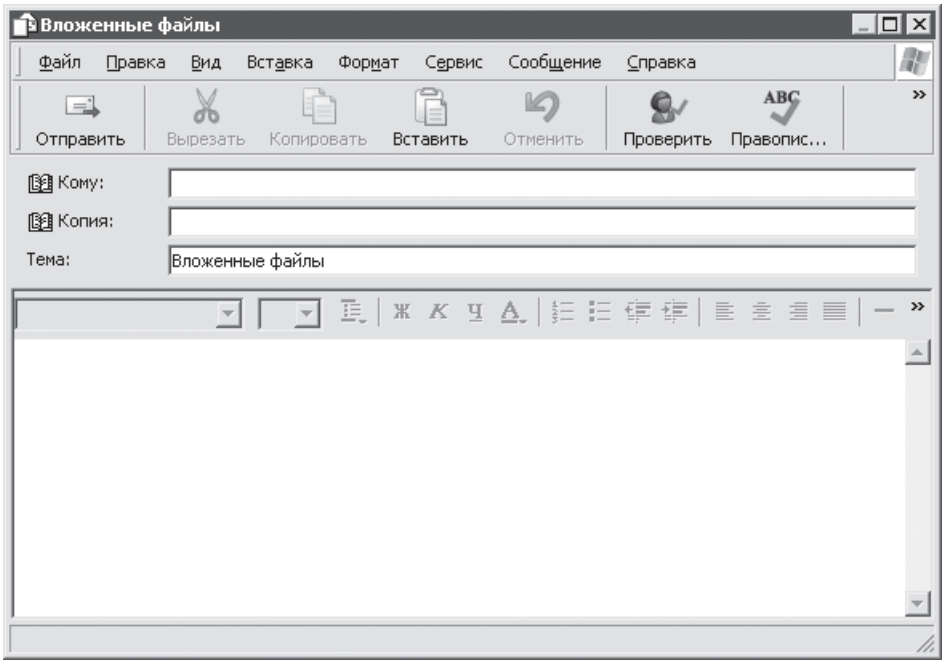


Рис. 3.3. Результат выполнения команды rundll32.exe "%programfiles%\Outlook Express\MSOE.DLL", MAPISendDocuments

Конфигурация

Теперь поговорим о конфигурации программы Outlook Express. В данном случае речь пойдет о различных компонентах почтового клиента, которые можно удалить либо воссоздать с помощью команд `rundll32.exe`.

Бывают ситуации, когда после неправильно установленной программы или сбоя системы перестают читаться файлы с различными расширениями. В нашем случае это файлы, предназначенные для открытия ссылок на почтовые или новостные серверы. Это говорит о том, что при сбое была удалена часть содержимого корневого раздела реестра `HKEY_CLASSES_ROOT` (о содержимом этого раздела читайте в следующей части) или только сведения о конкретных расширениях. Если это случилось и при этом не читаются только файлы, предназначенные для хранения ссылок на почтовые или новостные серверы, то нет смысла полностью переустанавливать почтовый клиент Outlook Express — намного проще воспользоваться несколькими командами `rundll32.exe`. Первой такой командой является следующая: `rundll32.exe "%programfiles%\Outlook Express\MSOE.DLL", SetDefaultMailHandler`. Она предназначена для переустановки всех сведений реестра об идентификаторе `mailto` (ветвь реестра `HKEY_CLASSES_ROOT\mailto`), который определяет файлы, предназначенные для описания ссылок на почтовые серверы. После выполнения данной команды идентификатор `mailto` будет удален из реестра и заново создан на основе стандартных настроек почтового клиента Outlook Express.

Второй такой командой является следующая: `rundll32.exe "%programfiles%\Outlook Express\MSOE.DLL", SetDefaultNewsHandler`. Она удаляет и заново восстанавливает по умолчанию содержимое идентификаторов `news`, `snews` и `nntp`.

ПРИМЕЧАНИЕ

Стоит отметить, что при использовании предыдущих команд будут созданы стандартные идентификаторы. Другими словами, они будут ссылаться на почтовый клиент Outlook Express, а не на клиент электронной почты, который установлен в данный момент в качестве клиента по умолчанию.

Еще одной проблемой, которая может произойти, является повреждение сведений об ActiveX-объектах, принадлежащих почтовому клиенту Outlook Express. Если эти повреждения незначительны, то можно попробовать исправить и их. Например, можно переписать сведения о доступных форматах импорта сообщений в почтовом клиенте. Эти сведения необходимы для работы Мастера импорта сообщений (Файл ▸ Импорт ▸ Сообщения) и позволяют импортировать сообщения формата Microsoft Exchange, Messenger, Netscape, Microsoft Mali и т. д. Если эти сведения будут повреждены, то единственным доступным форматом импорта будет формат Microsoft Outlook 6.0 (рис. 3.4) и, чтобы решить возникшую проблему, достаточно воспользоваться командой `rundll32.exe "%programfiles%\Outlook Express\oeimport.dll", DllRegisterServer`.

ПРИМЕЧАНИЕ

Возможна и обратная ситуация — когда вы намеренно хотите избавиться от ненужных форматов импорта/экспорта сообщений. В этом случае достаточно вос-

пользоваться командой rundll32.exe "%programfiles%\Outlook Express\oeimport.dll", DllUnregisterServer, и все сведения об импорте/экспорте сообщений исчезнут. Потом вы в любой момент сможете добавить эти сведения, если воспользуетесь описанной функцией библиотеки DllRegisterServer.

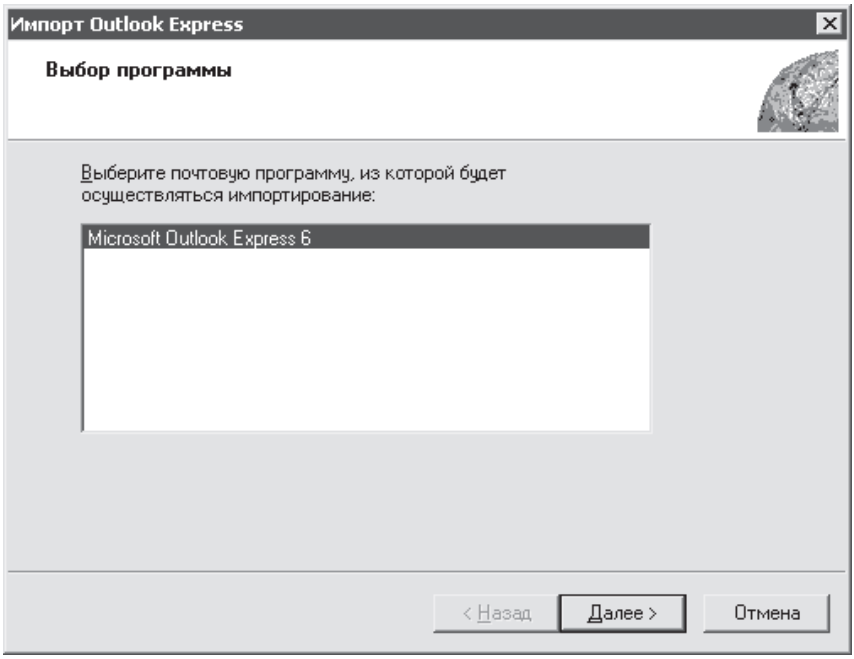


Рис. 3.4. Результат выполнения команды rundll32.exe "%programfiles%\Outlook Express\oeimport.dll", DllUnregisterServer

Но не только импорт сообщений можно восстановить — можно также восстановить диалоги импорта и экспорта адресных книг (Файл ▸ Импорт ▸ Другая адресная книга и Файл ▸ Экспорт ▸ Адресная книга). Если ActiveX-объекты для данных диалогов будут повреждены, то диалоги все равно будут отображаться, но воспользоваться ими будет нельзя. Чтобы восстановить ActiveX-объекты для импорта/экспорта адресных книг, необходимо вызвать следующую команду `rundll32.exe "%programfiles%\Outlook Express\WABIMP.dll", DllRegisterServer`. Кроме того, команды `rundll32.exe` также позволяют выполнить и противоположную операцию — удаление возможности использования импорта/экспорта адресных книг. Для этого достаточно применить команду `rundll32.exe "%programfiles%\Outlook Express\WABIMP.dll", DllUnregisterServer`.

Еще одной возможностью, которую можно восстановить, является возможность поиска людей в Интернете. Если поиск вам срочно необходим, то достаточно воспользоваться командой `rundll32.exe "%programfiles%\Outlook Express\WABfind.dll", DllRegisterServer`, и в подменю Найти меню Пуск появится команда Людей. Если же она там уже имеется, но вы ею никогда не пользовались

и пользоваться не собираетесь, то можно ее удалить. Для этого достаточно выполнить команду `rundll32.exe "%programfiles%\Outlook Express\WABfind.dll", DllUnregisterServer`.

Другие программы

Однако не только Internet Explorer и Outlook Express имеют в своих библиотеках функции, поддерживаемые командой `rundll32.exe`. Этим могут похвастаться и другие программы.

Программа Microsoft Visual Studio .NET является средой разработки, поддерживающей создание приложений на многих языках программирования. Неудивительно, что этот «тяжелый» по своей функциональности пакет поставляется вместе со многими библиотеками, которые можно использовать при работе с командой `rundll32.exe`.

Например, с помощью следующей команды: `rundll32.exe dfshim.dll, ShBackgroundUpdateW` — можно запустить службу фонового обновления программы Microsoft Visual Studio .NET. После выполнения данной команды будет запущен новый процесс `dfsvc.exe`.

Существует возможность прекращения работы данного процесса с помощью команды `rundll32.exe`. Для этого применяется следующая команда: `rundll32.exe dfshim.dll, KillService`.

Программа WMvare является эмулятором операционных систем семейств Linux, UNIX, Windows и т. д. Она включает в себя очень много команд `rundll32.exe`, некоторые из них уже были рассмотрены ранее (например, команды отключения и включения автозапуска компакт-диска). Перечислим другие команды этой программы.

- `rundll32.exe MSI39.dll, VMCleanFiles` — позволяет очистить компьютер от драйверов программы WMvare. После ее вызова происходит удаление из каталога `%systemroot%\SYSTEM32\DRIVERS` файлов драйверов `wmnetuserif.sys`, `wmnetbridge.sys`, `wmnet.sys` и `wmnetadapter.sys`.
- `rundll32.exe MSI39.dll, VMDeleteFiles` — если предыдущая команда позволяла удалять файлы драйверов WMvare, то эта команда дает возможность удалять остальные файлы WMvare, а также сведения о них из реестра.
- `rundll32.exe MSI39.dll, VMDeleteRegistry` — после вызова данной команды происходит удаление из реестра всех сведений о программе WMvare.
- `rundll32.exe MSI39.dll, VMCreateVMwareAccount` — позволяет создать учетную запись администратора WMvare.
- `rundll32.exe xvidvfw.dll, Configure` — с помощью этой команды можно вызвать диалог настройки Xvid (рис. 3.5).

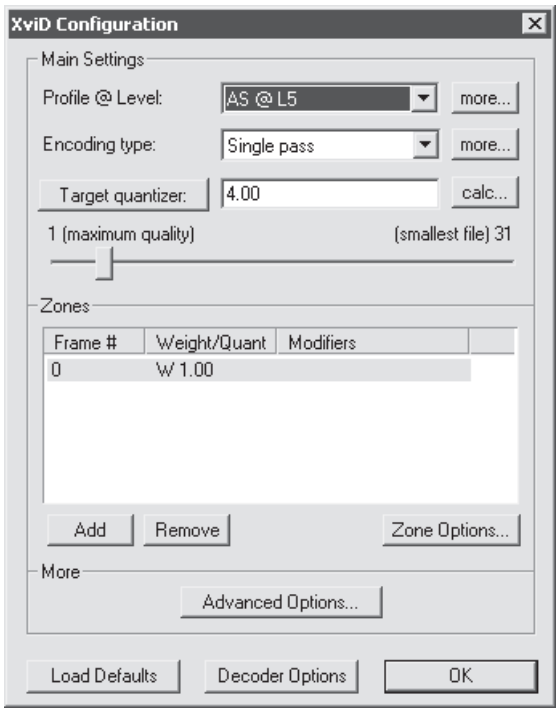


Рис. 3.5. Результат выполнения команды rundll32.exe xvidvfw.dll, Configure

Таким образом, в этой главе были рассмотрены некоторые интересные команды, используемые различными приложениями.

Часть 2

Реестр Windows XP

Глава 4

Корневой раздел HKEY_CLASSES_ROOT

- **Расширения файлов**
- **Подразделы корневого раздела**

В данной главе будет рассмотрен корневой раздел реестра HKEY_CLASSES_ROOT. Начнем с общих сведений о реестре.

Реестр Windows — это большая база данных, хранящая сведения обо всех настройках операционной системы. Будь то настройка цвета окна или пароль пользователя для входа в систему — все, вплоть до самого маленького флажка самого маленького диалогового окна, находится в реестре Windows.

Так уж получилось, что в Windows XP реестр реализован в виде набора файлов, доступ к которым можно получить только программным путем. Здесь не будет описываться содержимое этих файлов — это не является главной темой книги. Не будет описана и работа с программами для доступа к реестру — данная часть предназначена для тех, кто уже имеет некоторый опыт в использовании реестра. Но, несмотря на это, хотелось бы привести некоторую информацию для тех, кто уже забыл, как работать с реестром (но имеет опыт работы с ним — иначе стоит купить отдельную книгу, посвященную именно этой теме).

Для доступа к реестру в Windows XP применяется стандартная программа операционной системы regedit.exe, расположенная в каталоге %systemroot%. После ее запуска перед вами отобразится окно, состоящее из двух областей — в левой области отображаются ветви реестра, а в правой — содержащиеся в них параметры и их значения. Ветви реестра не произрастают из ничего, они имеют определенную точку монтирования — папку Мой компьютер. При этом стоит сказать, что от точки монтирования отходит пять ветвей (ни больше, ни меньше), называемых корневыми разделами и делящих содержащуюся в реестре информацию на определенные категории. Кратко опишем эти корневые разделы.

- HKEY_CLASSES_ROOT — содержит сведения обо всех расширениях файлов и ActiveX-объектах, зарегистрированных в системе. Глава 4 посвящена содержимому этого раздела.
- HKEY_CURRENT_USER — здесь находятся сведения о параметрах настройки оболочки Windows и конкретных установленных в ней программ для пользователя, работающего в данный момент с компьютером.
- HKEY_USERS — сведения для построения предыдущего раздела хранятся именно в этом корневом разделе. Раздел HKEY_USERS содержит данные о настройках оболочки Windows, применяемых для пользователя, впервые вошедшего в систему (в разделе .DEFAULT данного корневого раздела), а также настройки определенных классов пользователей и текущих пользователей системы. Другими словами, если в Windows 9x данный корневой раздел хранил сведения о настройках всех пользователей системы, то в Windows XP он хранит настройки только текущих пользователей, зарегистрированных в системе. Но если войти в программу regedit.exe от имени другого пользователя, то данный корневой раздел будет содержать настройки как пользователя, который сейчас зарегистрирован в системе, так и пользователя, от чьего имени был произведен запуск программы.

- HKEY_LOCAL_MACHINE — если предыдущие два корневых раздела включали в себя сведения о настройках определенных пользователей, то данный корневой раздел содержит информацию о настройках системы и программ, применяемых для всех пользователей системы.
- HKEY_CURRENT_CONFIG — корневой раздел включает в себя копию содержимого ветви реестра HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Hardware Profiles\Current и создается для совместимости с предыдущими версиями операционной системы.

Как уже было сказано, правая панель редактора реестра хранит сведения о содержащихся в открытой на данный момент ветви реестра параметрах: их названия, тип и значение. Параметры реестра могут иметь следующие типы данных (имеется в виду, что редактор реестра может создавать эти типы параметров, на самом же деле типов параметров намного больше).

- REG_SZ — строковый параметр, содержимым которого может быть строка символов Unicode.
- REG_EXPAND_SZ — расширяемый строковый параметр. Его содержимым может быть как произвольный текст, так и определенные переменные, которые после обработки системой преобразуются в статические пути к каталогам Windows (например, переменная %systemroot% преобразуется в каталог, в котором установлена текущая версия Windows, а переменная %username% преобразуется в имя текущего пользователя, работающего в системе).
- REG_MULTI_SZ — мультистроковый параметр, содержимым которого могут быть несколько строк текста, разграниченных NULL.
- REG_DWORD — если все предыдущие типы параметров хранили строковые данные, то данный тип определяет числовое значение длиной не более 4 байт.
- REG_BINARY — этот параметр хранит данные любой длины, но в основном применяется длина в 4 байта (по аналогии с REG_DWORD).

ПРИМЕЧАНИЕ

В основном нет никакой разницы между типами параметров реестра, поэтому вместо REG_DWORD-типа можно создавать параметры REG_BINARY-типа, вместо REG_SZ-типа можно создавать параметры REG_EXPAND_SZ-типа и т. д.

Битовая маска — это название, используемое для описания значения параметров типа REG_DWORD и REG_BINARY, при котором отдельно описывается результат установки каждого бита данного параметра. Например, в этой книге можно встретить абзацы, в которых будет содержаться подобный текст:

0□00000001 – □□□□□□□1
0□00000002 – □□□□□□□2

0□00000004 – □□□□□□□□3
0□00000008 – □□□□□□□□4
0□00000010 – □□□□□□□□5
0□00000020 – □□□□□□□□6
...

Приведенный абзац можно расшифровать так: если первый бит параметра будет установлен, то îïèñàíèå1, если второй бит параметра установлен, то îïèñàíèå2, если третий бит параметра установлен — îïèñàíèå3 и т. д.

При этом следует еще сказать, что в параметре может быть установлено сразу несколько битов, в этом случае выполняемое им действие определяется суммой соответствующих описаний. Битовая маска не только упрощает описание сложных параметров (по аналогии со способом их описания эти параметры будут также называться битовыми масками), но и позволяет легко установить отдельные биты параметра. Например, чтобы установить все приведенные выше биты параметра, нужно записать в него значение, равное 1 + 2 + 4 + 8 + 10 + 20 = 7 + 38 = F (расчет ведется в шестнадцатеричной системе счисления).

Рассмотрим наиболее интересные параметры реестра. Первым корневым разделом, структура которого будет описана, станет раздел HKEY_CLASSES_ROOT. Хотя, если быть точным, содержимое этого раздела строится на основе двух ветвей других корневых разделов: HKEY_LOCAL_MACHINE\SOFTWARE\Classes и HKEY_CURRENT_USER\Software\Classes. Первая ветвь реестра включает в себя информацию о расширениях файлов, используемую всеми пользователями компьютера построения карты расширений для . Вторая ветвь содержит информацию о расширениях, которые либо используются только текущим пользователем, либо переопределяют собой информацию из ветви корневого раздела HKEY_LOCAL_MACHINE. Иначе говоря, если информация о каком-нибудь расширении содержится как в корневом разделе HKEY_LOCAL_MACHINE, так и в HKEY_CURRENT_USER, то сведения из этих ветвей будут объединяться. При этом сведения из ветви HKEY_CURRENT_USER будут заменять собой сведения из ветви HKEY_LOCAL_MACHINE.

Корневой раздел HKEY_CLASSES_ROOT имеет более-менее статичную структуру. Если остальные корневые разделы могут хранить любую информацию, какую только захотят поместить в них программисты (любое название параметра и любой путь, ведущий к этому параметру), то HKEY_CLASSES_ROOT содержит параметры, названия которых не зависят от прихотей программистов, разрабатывающих соответствующее расширение файла или ActiveX-объект.

Как уже говорилось выше, корневой раздел HKEY_CLASSES_ROOT включает в себя всю информацию о расширениях файлов, которые зарегистрированы в вашей системе, а также описывает те действия, которые вы можете выполнить с файлами данного расширения (команды контекстного меню). Ветвь также содержит сведения обо всех зарегистрированных в системе ActiveX-компонентах, но о них будет

рассказано чуть позже, а сейчас разберемся с хранением сведений о расширениях файлов.

Расширения файлов

Есть два способа хранения в реестре информации о расширении файлов — правильный и не очень. Второй применялся в старых версиях операционной системы Windows, хотя поддерживается и Windows XP. При его использовании все параметры и дочерние подразделы записываются в один раздел HKEY_CLASSES_ROOT. Таким образом, все то, что будет рассмотрено далее в этом разделе, при использовании неправильного способа хранения будет содержаться в одном подразделе корневого раздела HKEY_CLASSES_ROOT. Правильный же способ хранения параметров расширения файла определяет для их хранения два раздела. Первый из них назван в честь расширения файла, например, для файлов с расширением TXT он будет называться .txt. Этот раздел еще называют разделом расширения. Он практически ничего интересного не содержит, но зато в параметре (Ïî óìîë÷àíèþ) находится название второго раздела. Вот этот раздел реестра как раз и включает в себя всю интересную и увлекательную информацию о расширении. Он еще называется разделом идентификатора.

В книге будут описываться параметры и дочерние разделы, используемые в регистрации расширения с помощью правильного способа. Но сначала посмотрите на рис. 4.1. Он иллюстрирует правильный способ хранения информации для файлов с расширением TXT.

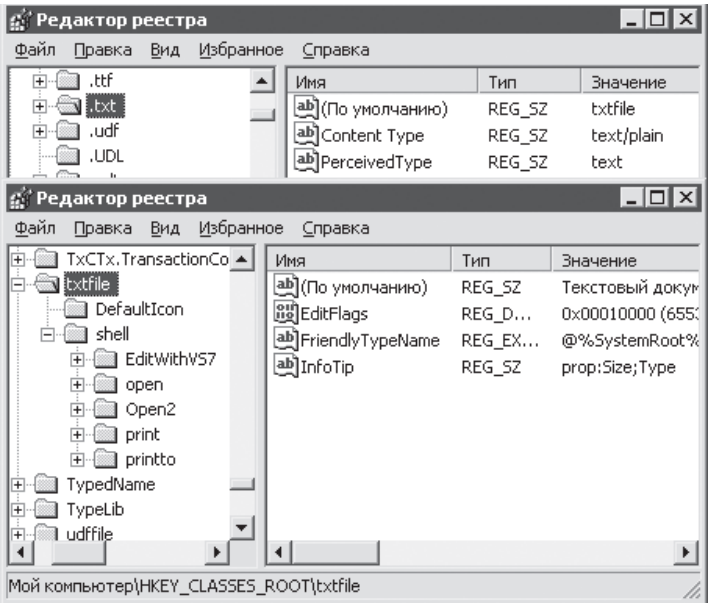


Рис. 4.1. Разделы описания расширения файла

Раздел расширения

Как уже говорилось, раздел расширения включает в себя параметр (Ïî óìîë÷àíèþ), значение которого определяет название раздела идентификатора, описывающего данное расширение. Но, кроме этого параметра, раздел расширения может содержать еще несколько параметров строкового типа.

- Content Type — определяет тип расширения MIME, который ассоциирован с файлами соответствующего расширения. Те, кто занимался веб-программированием, конечно знают, что такое MIME. Именно строка MIME используется браузером для определения того, что же ему нужно делать с соответствующим расширением (то ли открыть, то ли воспроизвести, то ли еще что-нибудь).

  На рис. 4.1 видно, что текстовые файлы (TXT) используют расширение text/plain, которое определяет содержимое данных файлов как неформатированный текст.

- PerceivedType — говорит системе о том, в каком виде хранится информация с соответствующим расширением. Этот параметр может принимать такие значения: Text, Image, Audio, Video, System, Compressed.

- Generic — если значение равно System, то файлы с соответствующим расширением являются собственностью системы и сведения о них лучше не удалять.

- NoOpen — применяется только в том случае, когда параметр (Ïî óìîë÷àíèþ) раздела расширения не имеет никакого значения. Если данный параметр присутствует, то при попытке открытия соответствующего файла перед отображением диалога Открыть с помощью будет выводиться диалог Внимание, с сообщением о том, что файлы с данным расширением используются системой и их лучше не открывать.

Данный раздел может содержать подразделы. Например, в нем может храниться подраздел ShellNew, который определяет команды в списке Создать контекстного меню Рабочего стола или Проводника. Иначе говоря, если раздел расширения включает в себя этот подраздел (при этом в нем должен находиться один из описанных ниже параметров), то в списке Создать появится пункт, с помощью которого можно будет создать файл с соответствующим расширением.

Как сказано выше, подраздел ShellNew должен содержать определенный параметр. Этот параметр как раз и определяет, что именно система должна сделать после того, как пользователь выбрал соответствующую команду меню Создать. Рассмотрим возможные параметры (в подразделе ShellNew должен находиться только один из этих параметров).

- Command — этот параметр строкового типа определяет команду, которая будет выполняться при выборе соответствующего пункта меню Создать.

- NullFile — данный строковый параметр вообще не должен содержать никакого значения (при этом после выбора создания соответствующего расширения будет создаваться пустой файл).

- `FileName` — этот параметр строкового типа определяет путь и имя файла, который будет создаваться (просто копироваться в текущую папку) после выбора соответствующей команды меню Создать. При этом если файл находится в каталоге, определяемом содержимым параметра строкового типа `Templates`, расположенного в ветви реестра `HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\Shell Folders`, то достаточно присвоить параметру `FileName` только имя необходимого файла.
- `Data` — параметр `BINARY`-типа представляет собой последовательность цифр в шестнадцатеричной системе счисления, которую система будет применять для генерации содержимого файла (например, этот параметр используют файлы с музыкальным контекстом для возможности создания пустого файла-образца звука).

Для примера попробуем создать свою собственную команду в списке Создать. Поскольку большая часть книги посвящена описанию работы с сервером сценариев Windows, сначала будет описан шаблон для быстрого создания основы файла сценария VBS. Для этого нужно воспользоваться ветвью реестра `HKEY_CLASSES_ROOT\.vbs\ShellNew`. Необходимо создать в ней расширяемый строковый параметр `FileName`, которому нужно присвоить путь к файлу шаблона. Поскольку в шаблоне будут строки вызова объектов Windows, другие параметры данной ветви не подойдут. Например, присвойте данному параметру значение `%systemroot%\WSHtemplate.vbs`. Файл шаблона должен содержать следующие данные:

```
set wshshell = WScript.CreateObject("WScript.Shell")
```

Пока хватит и этого вызова, когда вы займетесь непосредственно созданиями сценариев, то модифицируете файл шаблона. Теперь присвойте созданному файлу имя `WSHtemplate.vbs` и посмотрите на результат (рис. 4.2).

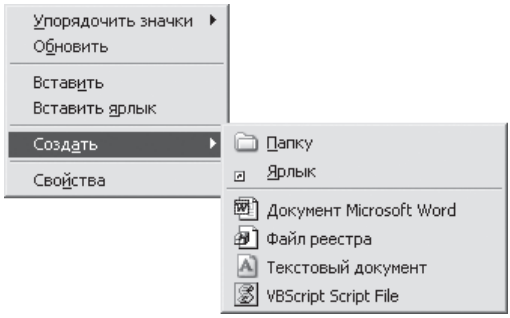


Рис. 4.2. Результат создания команды контекстного меню

Раздел идентификатора

Теперь поговорим о содержимом второго раздела, используемого для описания расширения.

Кроме разделов идентификаторов для расширений файлов, в системе также существуют стандартные идентификаторы Windows, описывающие такие объекты, как папка, диск, неизвестные расширения.

- `Folder` — содержит настройки для папок Windows. Команды контекстного меню, описываемые данным идентификатором, могут использоваться в построении контекстного меню таких объектов, как диски, директории, значки Корзина и Мой компьютер.
- `Directory` — определяет настройки контекстного меню для директорий Windows и содержит подраздел `Background`. Содержимое этого подраздела определяет команды Упорядочить значки и Создать для контекстного меню Рабочего стола и Проводника.
- `Drive` — содержит настройки значков дисков в системе. При этом системой могут применяться и дополнительные идентификаторы для отображения дисков, например, если в данный момент в привод компакт-дисков вставлен музыкальный диск, то будет использоваться идентификатор AudioCD.
- `CompressedFolder` — определяет настройки сжатых папок.
- `*` — если все предыдущие идентификаторы описывали настройки конкретных файлов расширений, то данный идентификатор определяет настройки для всех расширений файлов, зарегистрированных в системе. Именно сюда довольно часто добавляют свои команды контекстного меню различные программы. Например, идентификатор используется программами WinRAR и «Антивирус Касперского» для добавления своих команд в контекстное меню файлов.
- `Unknown` — определяет настройки для всех файлов, незарегистрированных в системе. Обычно он описывает только команду Открыть с помощью.
- `AllFilesystemObjects` — является родительским для всех других идентификаторов. Он определяет настройки для всего: для зарегистрированных файлов, для незарегистрированных файлов, для папок, дисков и т. д. По умолчанию данный идентификатор определяет только команду Отправить.

Параметры раздела идентификатора

Раздел идентификатора может содержать следующие параметры.

- `EditFlags` — данный параметр `DWORD`-типа определяет различные ограничения на модификацию параметров данного расширения при помощи вкладки Типы файлов диалога Свойства папки. Например, если значение равно `0õ00000001`, то соответствующего расширения в списке зарегистрированных файлов данной вкладки вы не найдете. Если же значение равно `0õ00000008`, то кнопка Дополнительно вкладки Типы файлов для данного расширения будет заблокирована. Если же значение равно `0õ00000200`, то будет запрещена возможность смены значка для данного расширения.

  Можно вообще сложить все предыдущие значения — параметр является набором управляющих битов (битовой маской) — это приведет к одновременному скрытию типа файлов из диалога Свойства папки, блокировке кнопки Дополнительно для данного расширения и запрету изменения значка.

Для примера на рис. 4.3 приведен диалог изменения TXT-файла с установленным значением параметра в 0õ000003ñ0, то есть используются следующие битовые маски:

- 0õ00000040 — делает недоступной кнопку Изменить диалога Изменение свойств типа файлов (диалог вызывается при нажатии кнопки Дополнительно на вкладке Типы файлов);
- 0õ00000080 — делает недоступной кнопку Удалить диалога Изменение свойств типа файлов;
- 0õ00000100 — запрещает пользователю изменение описания файлов с данным расширением при помощи поля диалога Изменение свойств типа файлов;
- 0õ00000200 — запрещает пользователю изменение пиктограммы, присвоенной файлам с данным расширением, при помощи кнопки Сменить значок в окне Изменение свойств типа файлов.

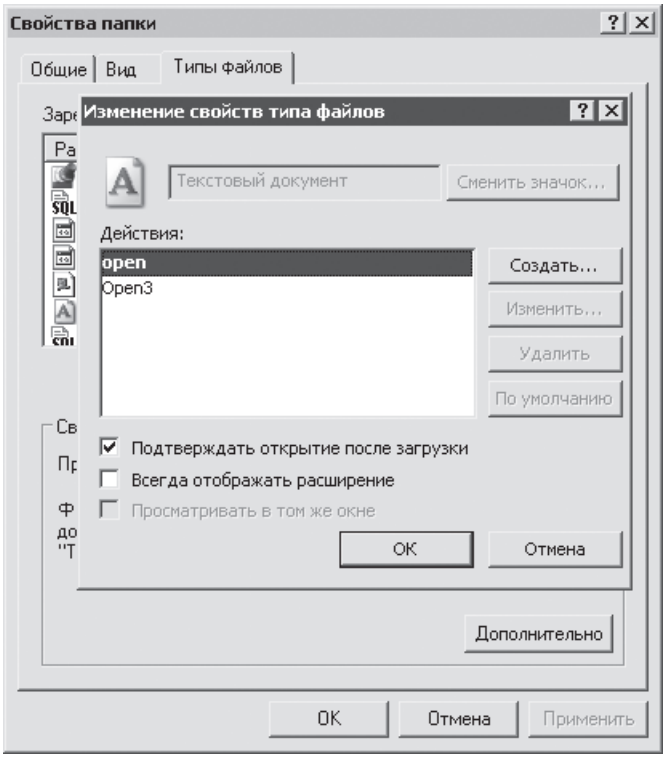


Рис. 4.3. Запрет на удаление, изменение, смену описания и смену значка

- AlwaysShowExt и NeverShowExt — два эти параметра строкового типа являются взаимоисключающими, то есть если в разделе присутствует один из них, то другого там быть не должно (значения им присваивать не нужно). Если в разделе будет присутствовать первый параметр, то соответствующее расширение

файла всегда будет отображаться. Независимо от того, как настроена система (здесь имеется в виду возможность скрытия зарегистрированных расширений при помощи соответствующего флажка вкладки Вид диалога Свойства папки). Если присутствует второй параметр, то расширение всегда будет скрыто.

- `InfoTip` — этот параметр строкового типа определяет подсказку, которая будет отображаться при удержании указателя мыши над файлом данного расширения. Параметр может содержать или произвольный текст, или специальные зарезервированные слова (если присутствуют обычные слова, то зарезервированные также считаются обычными и не выполняют заложенные в них действия). Зарезервированные слова пишутся после строки `prop:`. К таким словам относятся следующие.
  - `Comment` — в подсказке будет выводиться комментарий, вводимый в диалоге Свойства данного файла.
  - `Size` — отображается размер файла.
  - `Access` — отображаются права на доступ к файлу.
  - `Owner` — отображается логин создателя файла.
  - `Year` и другие. Несколько зарезервированных слов пишутся через точку с запятой.

  На рис. 4.1 можно увидеть, что для TXT-файлов в подсказке отображаются сведения о размере, а также описание файла, задаваемое в параметре `(Ïî óìîë÷àíèþ)` раздела идентификатора.

  Параметр `(Ïî óìîë÷àíèþ)` раздела идентификатора содержит строку описания, отображаемую в поле Тип файла диалога Свойства. Значение данного параметра также отображается как название команды в пункте меню Создать контекстного меню Рабочего стола или Проводника.

- `IsShortCut` — это необязательный параметр строкового типа, не содержащий никакого значения. Если он присутствует в разделе, то файлы с соответствующим расширением будут считаться ярлыками. Это приведет к тому, что внизу значка файла будет появляться стрелка, символизирующая файл ярлыка.

- `BrowseInPlace` — очень интересный строковый параметр, значение которого можно не указывать. Он используется только для идентификаторов, файлы которых открывают окно Проводника — например, для идентификатора `Directory`. Если данный идентификатор будет содержать параметр строкового типа `BrowseInPlace`, то все папки в системе будут открываться с помощью браузера Internet Explorer. Например, если попытаться открыть папку `C:\Windows`, то браузер будет искать сайт с адресом C:\Windows. Конечно, у него это не получится, поэтому с помощью данного параметра строкового типа можно запретить пользователям доступ к папкам системы.

- `DocObject` — этот параметр строкового типа также очень интересен, ведь если его создать, допустим, в разделе `Directory`, то можно добиться таких же действий, что и с помощью предыдущего параметра.

- `Thumbnail` — еще один параметр строкового типа. Его значение указывает на файл изображения, который будет использоваться для отображения в виде

эскиза страницы для файлов с соответствующим расширением (правда, данный параметр применяется не для всех идентификаторов). Например, если создать данный параметр в разделе идентификатора `Drive`, то все диски при использовании вида Эскизы страниц будут представлены с помощью указанного вами изображения. Если же создать данный параметр в разделе идентификатора `Folder`, то ваш значок будет использоваться для отображения значка Корзины.

- `DefaultDropEffect` — данный параметр `DWORD`-типа определяет, какую именно операцию будет выполнять система при перемещении в другое окно файлов с данным расширением. По умолчанию при этом она перемещает файл любого расширения в соответствующую папку, но если значение этого параметра равно `1`, то файл с соответствующим расширением будет копироваться.

Разделы ветви идентификатора

Раздел идентификатора, кроме параметров, может включать в себя и другие разделы — они определяют значок, используемый для файлов с данным расширением, текущую версию идентификатора, а также сами команды контекстного меню файла. Рассмотрим содержимое этих разделов.

- `CurVer` — параметр `(Ïî óìîë÷àíèþ)` данного раздела определяет названия идентификатора, имеющего более новые сведения о расширении. Если вы встретите этот раздел в каком-нибудь идентификаторе расширения, то можете сразу смотреть на его параметр `(Ïî óìîë÷àíèþ)` и искать записанный в нем идентификатор — ведь настройки текущего идентификатора в системе уже не применяются.
- `DefaultIcon` — параметр `(Ïî óìîë÷àíèþ)` этого раздела определяет путь к BMP-файлу изображения, применяемому для отображения значков файлов с соответствующим расширением.
- `Shell` — это, наверное, самый интересный раздел идентификатора, ведь именно его содержимое и определяет команды контекстного меню для расширения, а также их свойства. Параметр `(Ïî óìîë÷àíèþ)` данного раздела определяет название глагола (глаголы являются подразделами раздела `shell`), команда которого будет использоваться при выполнении попытки открытия файлов соответствующего расширения двойным щелчком кнопкой мыши. Параметр `(Ïî óìîë÷àíèþ)` может также хранить перечисление всех глаголов раздела — в этом случае он описывает последовательность, в которой они будут расположены в контекстном меню файла.

Сами же глаголы, как уже было сказано, являются подразделами ветви `shell` и могут содержать различные параметры (один глагол определяет одну команду контекстного меню). Параметр `(Ïî óìîë÷àíèþ)` глагола определяет название команды, которая будет идентифицировать соответствующее действие в контекстном меню файлов. Если данный параметр не будет определен, то в качестве названия команды будет использоваться название подраздела глагола.

Данное правило не распространяется на так называемые стандартные глаголы системы — если их параметр `(Ïî óìîë÷àíèþ)` не определен, то будет использоваться

название команды контекстного меню, заложенное в системе для данных глаголов. К стандартным можно отнести следующие глаголы:

- `Open` — добавляет в контекстное меню команду Открыть;
- `Explore` — Проводник;
- `Find` — Найти;
- `Openas` — Открыть с помощью;
- `Runas` — Запуск от имени;
- `Print` — Печать;
- `Printo` — в отличие от предыдущих этот глагол не создает команды контекстного меню, зато он добавляет возможность перетаскивания файлов на значок принтера.

Глаголы могут содержать следующие параметры.

- `MUIVerb` — если этот параметр строкового типа присутствует в системе, то его значение будет переопределять значение параметра `(Ïî óìîë÷àíèþ)` данного глагола.
- `FriendlyAppName` — еще один параметр строкового типа. Он переопределяет команду, отображаемую в списке Открыть с помощью и идентифицирующую программу, запускаемую с помощью данного глагола. Например, если создать этот параметр в ветви реестра `HKEY_CLASSES_ROOT\txtfile\shell\open` и присвоить ему значение, допустим, `Ìîé áëîêíîòè`, то можно будет увидеть диалог, представленный на рис. 4.4.

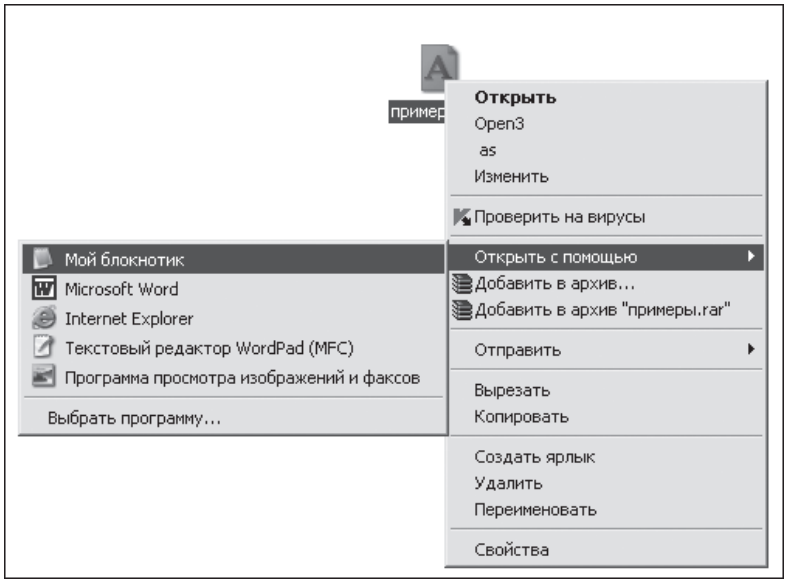


Рис. 4.4. Создание синонима программы

- BrowserFlags — этот параметр иногда можно встретить в ветви реестра HKEY_CLASSES_ROOT\Folder\shell\open. Он является битовой маской и может определять способ открытия папки и всех ее дочерних объектов в Windows. Например, если в значении этого параметра присутствует битовая маска 0õ00000020, то папки будут открываться с помощью Проводника, то есть с открытой панелью папок в левой части окна.
- Extended — если данный параметр строкового типа существует в подразделе глагола, то описываемая этим подразделом команда не будет отображаться в контекстном меню файлов соответствующего расширения.

Кроме параметров, подразделы глаголов включают в себя и несколько подразделов, которые как раз и определяют команды, выполняемые при выборе из контекстного меню файла соответствующего действия.

- Command — параметр (Ïî óìîë÷àíèþ) данного подраздела содержит команду, которая будет выполняться при выборе из контекстного меню файла соответствующего действия.

ПРИМЕЧАНИЕ

В реестре Windows XP существует ветвь, переопределяющая программу, которая будет запускаться при двойном щелчке кнопкой мыши на файле (то есть переопределяющая параметр (По умолчанию) подраздела command для глагола open). Этой ветвью является HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\FileExts\«расширение». Она может хранить строковый параметр Application, который как раз и определяет приложение, запускаемое при двойном щелчке на файле с соответствующим расширением.

- Ddeexec — подраздел необязателен. Его содержимое определяет команду DDE, которая будет исполняться вместе с указанной командой подраздела command при выборе из контекстного меню соответствующего действия. Само сообщение DDE описано в параметре (Ïî óìîë÷àíèþ) данного подраздела, а параметры для его исполнения записаны в дочерних подразделах. Более подробно об этих параметрах можно узнать из базы данных, поставляемой с книгой (в самой книге эти параметры рассмотрены не будут, так как на практике обычные пользователи их применяют не часто).

ПРИМЕЧАНИЕ

DDE (динамический обмен данными) — это специальные команды, которые позволяют нескольким работающим приложениям обмениваться между собой данными. Например, с помощью DDE можно из одного приложения управлять работой другого.

Но не только раздел shell может хранить настройки контекстного меню файлов с соответствующим расширением — команды контекстного меню могут содержаться

и в разделе shellex. Нет, это не избыточность — раздел shellex имеет совершенно не такое назначение, как shell. Если раздел shell создан специально для того, чтобы описывать команды меню, вызывающие различные программы с помощью стандартного диалога Запуск программы, то shellex хранит сведения о дополнительных расширениях, которые могут использоваться при отображении значка, контекстного меню или диалога свойств для файлов данного расширения. Эти расширения реализованы в системе с помощью ActiveX-объектов, но о них мы скажем чуть позже, когда будет рассматриваться ветвь реестра, содержащая сведения обо всех установленных на компьютере ActiveX-объектах.

В зависимости от назначения ActiveX-объекта его описание должно находиться в одном из дочерних подразделов раздела shellex. Например, в подразделе PropertySheetHandlers хранятся дополнительные вкладки для диалога Свойства файлов данного расширения. В подразделе ContextMenuHandlers определяются дополнительные команды контекстного меню, использующие для своей работы ActiveX-объекты. В подразделе DropHandler определяется обработчик для операций drag-and-drop, выполняемых с файлами соответствующего расширения, а в подразделе IconHandler находятся сведения об обработчиках значков, которые будут выводить соответствующий значок в диалоге свойств или вместо стандартного значка файла.

Независимо от того, какой подраздел используется для описания расширения ActiveX-файла, его содержимое должно быть представлено в одной из следующих форм.

- С помощью набора подразделов, названных в честь соответствующих CLSID-номеров ActiveX-объектов (один подраздел — описание одного расширения).
- С использованием подразделов, название которых не имеет значения. В этом случае параметры (Ïî óìîë÷àíèþ) данных подразделов должны хранить значения, названные в честь соответствующих CLSID-номеров ActiveX-объектов (один подраздел — описание одного расширения).

Подразделы корневого раздела

Как видите, сведения о расширениях файлов хранятся в реестре в виде упорядоченных данных (чего, к сожалению, нельзя сказать о других корневых разделах реестра, структура которых если и прослеживается, то только в некоторых местах).

Но не только сведения о расширениях файлов можно найти в корневом разделе HKEY_CLASSES_ROOT — в нем также хранятся сведения обо всех установленных на компьютере ActiveX-объектах и некоторые другие разделы, описание которых приведено далее.

CLSID

В данной главе книги уже несколько раз упоминалось о таких объектах Windows, как ActiveX, но еще ни слова не говорилось о том, что же это такое и для чего

необходимо. Но этот пробел будет восполнен, ведь сейчас начнется рассказ о структуре раздела CLSID, который включает в себя сведения об ActiveX-объектах.

ActiveX-объекты — это специальные, уже скомпилированные программы, доступ к которым нельзя получить напрямую, но их можно вызвать с помощью операционной системы Windows. Всем известными примерами ActiveX-объектов могут быть следующие значки: Мой компьютер, Мои документы, Корзина, Назначенные задания, Панель управления и т. д.

Каждый ActiveX-объект имеет свой уникальный идентификатор — CLSID-номер, который служит в системе в качестве названия ActiveX-объекта. CLSID-номера не берутся из головы — для их создания предназначены специальные программы. Например, программа GUIDgen, входящая в стандартную поставку компилятора Microsoft Visual C++ 6.0.

CLSID-номер является 32-байтным номером, состоящим из шестнадцатеричных чисел, первые восемь байт которого генерируются случайным образом. Следующие четыре байта используют для своего создания текущее значение даты и времени, а остальные генерируются на основе конфигурационных данных компьютера. При этом CLSID-номер берется в фигурные кавычки и имеет такой формат написания: {XXXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX}.

Перечень CLSID-номеров наиболее интересных ActiveX-объектов можно просмотреть в приложении 4.

Параметры раздела ActiveX-объекта

Раздел CLSID включает в себя список вложенных подразделов, каждый из которых назван на основе CLSID-номера ActiveX-объекта, который он описывает, и хранит сведения только об этом ActiveX-объекте. Подраздел ActiveX-объекта может содержать следующие параметры.

- (Ïî óìîë÷àíèþ) — определяет дружественное название ActiveX-объекта, предназначенное для отображения в качестве имени значка или команды. Например, если изменить значение параметра (Ïî óìîë÷àíèþ) для ActiveX-объекта {645FF040-5081-101B-9F08-00AA002F954E} (значок Корзины), то изменится и само название Корзины, отображаемое под ее значком. Если изменить значение параметра (Ïî óìîë÷àíèþ) ActiveX-объекта {21EC2020-3AEA-1069-A2DD-08002B30309D}, то изменится заголовок и адрес панели управления при ее отображении (рис. 4.5).

ПРИМЕЧАНИЕ

Если вам необходимо изменить имя сетевого клиента (по умолчанию это имя — Microsoft Windows Network, но можно изменить его на любое другое, например Моя сеть), которое можно найти по пути Сетевое окружение ▸ Вся сеть, то ActiveX-объект сетевого клиента не поможет. Данное имя считывается из параметра строкового типа Name ветви реестра HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\lanmanworkstation\NetworkProvider — именно этот параметр и нужно изменять.

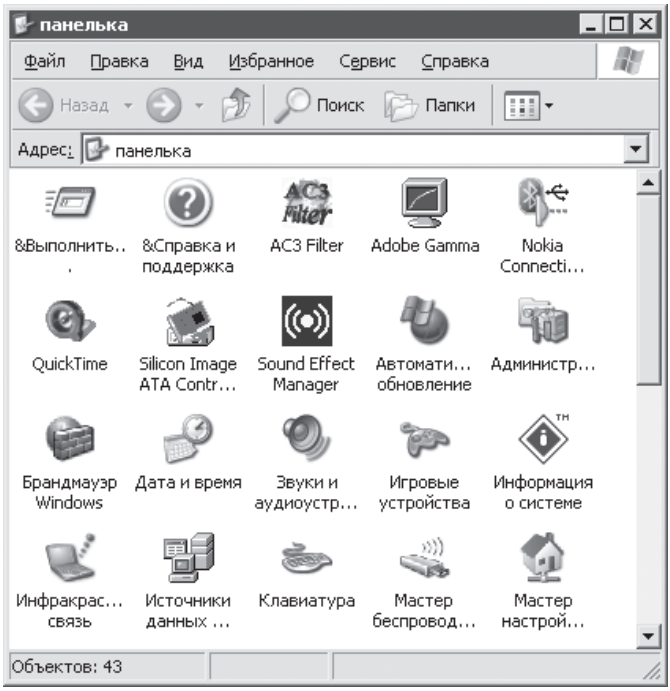


Рис. 4.5. Результат изменения названия Панели управления

- `LocalizedString` — используется в Windows для локализации операционной системы и содержит название ActiveX-объекта, заменяющее собой название из параметра (`Ïî óìîë÷àíèþ`).
- `SortOrderIndex` — этот параметр `DWORD`-типа может присутствовать в некоторых подразделах ActiveX-объектов, отображаемых на Рабочем столе, например в подразделах ActiveX-объектов Мой компьютер, Мои документы и Корзина. Он определяет порядок расположения этих ActiveX-объектов на Рабочем столе.
- `InfoTip` — данный параметр строкового типа определяет строку подсказки, отображаемую для значка соответствующего ActiveX-объекта при наведении и удержании на нем указателя мыши.

Подразделы ветви ActiveX-объекта

Подразделы, описывающие ActiveX-объекты, кроме параметров, могут содержать множество дочерних подразделов. Среди них могут присутствовать и уже рассмотренные `shell` и `shellex`, а также некоторые из следующих подразделов.

- `DefaultIcon` — параметр (`Ïî óìîë÷àíèþ`) данного подраздела определяет путь к файлу рисунка, применяемому для отображения значка ActiveX-объекта (например, этот параметр ActiveX-объекта `{20D04FE0-3AEA-1069-A2D8-08002B30309D}` определяет значок для папки Мой компьютер).
- Данный подраздел может также хранить параметр строкового типа `OpenIcon`, который определяет путь к файлу рисунка, используемому вместо стандартного

рисунка значка открытого ActiveX-объекта. На рис. 4.6 можно увидеть результат изменения этого параметра для приведенного выше ActiveX-объекта `{20D04FE0-3AEA-1069-A2D8-08002B30309D}`.

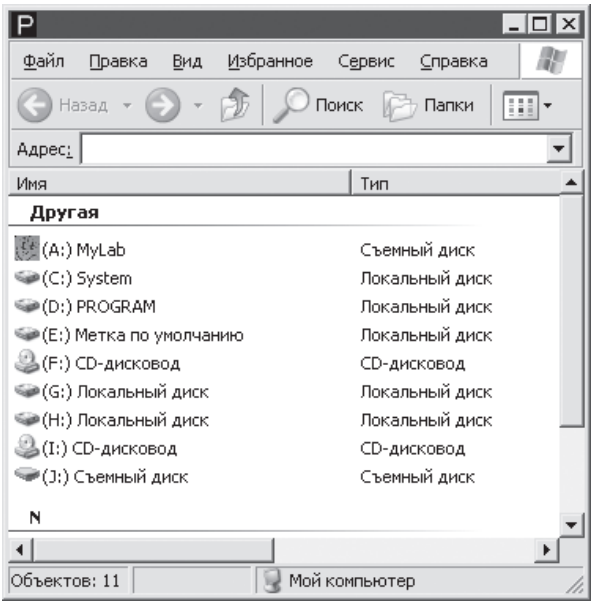


Рис. 4.6. Изменение значка на панели заголовка папки Мой компьютер

ПРИМЕЧАНИЕ

Значение приведенного выше параметра (а также некоторых других параметров, которые будут описаны) может быть переопределено в ветви реестра HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\CLSID\{CLSID-номер ActiveX-объекта}.

- `ShellFolder` — параметры этого подраздела определяют различные свойства значка ActiveX-объекта, а также команд его контекстного меню. Подраздел может включать в себя следующие параметры.
  - `WantsFORDISPLAY` — присутствие этого параметра строкового типа запрещает системе выводить название соответствующего ActiveX-объекта под его значком. Например, если создать этот параметр в ветви реестра `{645FF040-5081-101B-9F08-00AA002F954E}`, то можно добиться такого же эффекта, как и на рис. 4.7.

Рис. 4.7. Скрытие названия значка Корзины

- `HideInWebView` — присутствие этого параметра строкового типа говорит системе о том, что она должна скрывать значки, соответствующие данному ActiveX-объекту, при использовании нового вида окна Проводника (при установке переключателя в положение Отображение списка типичных задач в папках на вкладке Общие диалога Свойства папки).
- `HideFolderVerbs` — присутствие данного параметра строкового типа говорит системе о том, что она не должна добавлять к контекстному меню значка ActiveX-объекта команды контекстного меню стандартного идентификатора `Folder`.

ПРИМЕЧАНИЕ

Существует и альтернативная возможность — сказать системе, что она обязана добавить к контекстному меню данного ActiveX-объекта команды, определяемые идентификатором Folder. Для этого необходимо установить битовую маску 0x20000000 в параметре DWORD-типа Attributes, описание которого приведено ниже.

- `Attributes` — значение данного параметра `DWORD`-типа определяет, будут ли отображаться различные стандартные команды контекстного меню для значка данного ActiveX-объекта. Параметр является битовой маской, биты которой имеют следующую функциональность: `0õ00000001` — определяет присутствие команды Копировать; `0õ00000002` — устанавливает присутствие команды Вырезать; `0õ00000010` — определяет команду Переименовать; `0õ00000020` — устанавливает команду Удалить; `0õ00000040` — определяет отображение команды Свойства; `0õ00000100` — устанавливает отображение команды Вставить; `0õ00200000` — определяет отображение команд, содержащихся в подразделе `ContextMenuHandlers` (он является дочерним по отношению к разделу `shellex`, который, в свою очередь, располагается в одном из разделов идентификаторов).

Например, на рис. 4.8 отображено контекстное меню значка Корзины с использованием значения параметра `Attributes`, равного `0õ20000030`.

Рис. 4.8. Редактирование контекстного меню значка Корзины

ПРИМЕЧАНИЕ

В подразделе ShellFolder может присутствовать DWORD-параметр CallForAttributes. Если его значение отлично от 0, то система не будет использовать содержимое параметра Attributes для ограничения контекстного меню значка данного ActiveX-объекта.

Это далеко не все подразделы, которые могут находиться в ветви ActiveX-объекта, но это самые интересные из них (с точки зрения возможностей изменения). О других подразделах ветви ActiveX-объекта можно узнать в базе данных по реестру, поставляемой вместе с этой книгой.

Другие разделы корневого раздела

Уже была рассмотрена большая часть содержимого корневого раздела реестра HKEY_CLASSES_ROOT — разделы расширения, идентификатора и раздел CLSID, но, кроме них, корневой раздел включает в себя еще несколько разделов, которые стоит описать. Некоторые из описываемых в разделах параметров реализуют уникальные возможности, которых в Windows больше никакими другими способами достичь нельзя.

- .DeskLink — реализует возможность создания ярлыков на Рабочем столе с помощью команды контекстного меню файлов Отправить. Он описывает ActiveX-объект, предназначенный для выполнения этой операции. Если удалить или переименовать этот раздел, то соответствующая возможность будет запрещена.
- Applications — определяет список программ, которые будут отображаться в диалоге Открыть с помощью. Он включает в себя подразделы, названные в честь соответствующих программ. Если подраздел присутствует, то диалог Открыть с помощью будет содержать возможность открытия файла с помощью этой программы.

  Раздел Applications имеет и противоположную описанной выше функции возможность — он определяет те программы, которые никогда не будут отображаться в диалоге Открыть с помощью. Если программа не должна отображаться в диалоге, то в дочернем подразделе раздела Applications, названном в честь данной программы, нужно создать параметр строкового типа NoOpenWith. Значение этого параметра не используется.

  Можно еще создать строковый параметр NoStartPage. Если в дочернем подразделе раздела Applications, названном в честь данной программы, будет находиться приведенный строковой параметр, то, сколько бы вы ни вызывали соответствующую программу, она не будет отображаться в списке часто используемых программ нового меню Пуск.

  Еще один параметр строкового типа, который может содержаться в подразделе, названном в честь необходимой программы, — TaskbarGroupIcon. Его значение определяет путь к файлу рисунка, который будет использоваться в качестве значка сгруппированных программ на Панели задач. Например, можно создать данный параметр в ветви реестра HKEY_CLASSES_ROOT\Applications\explorer.exe (рис. 4.9).

  Но, кроме параметров, раздел Applications может хранить и другие подразделы. Например, в нем может находиться подраздел shell, включающий в себя дополнительные глаголы (или переопределение уже существующих) для файлов, ассоциированных с соответствующей программой уже после того, как будет изменено содержимое подраздела shell ветви HKEY_CLASSES_ROOT\Applications\«ïðèëîæåíèå».

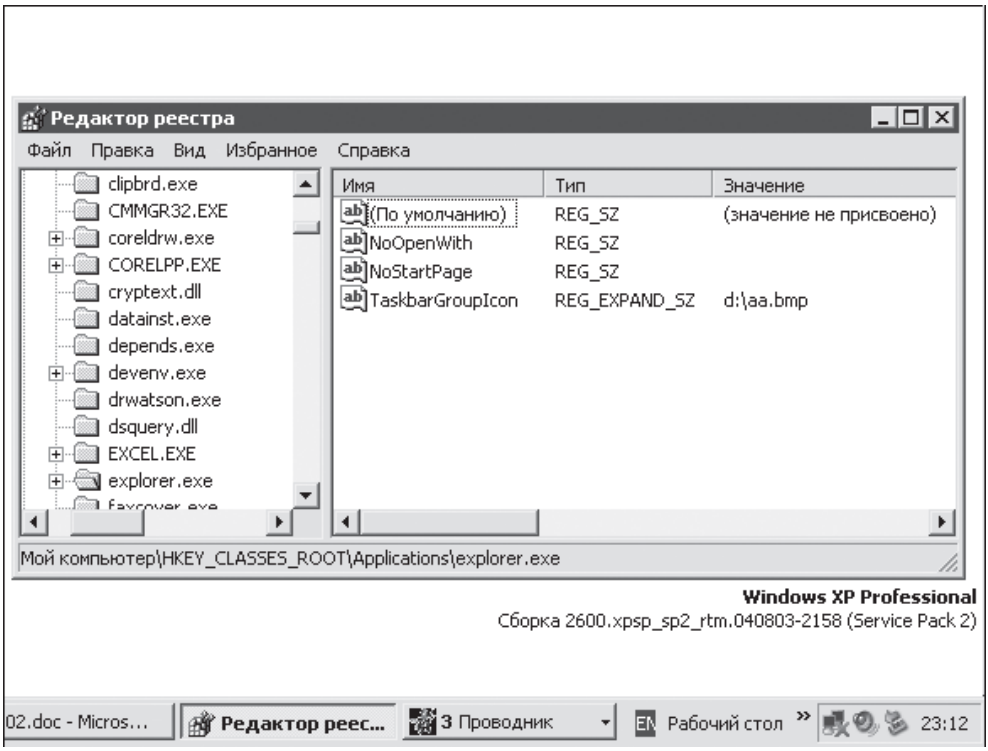


Рис. 4.9. Изменение рисунка группы программ

Кроме подраздела `shell`, в некоторых разделах ветви системного реестра `HKEY_CLASSES_ROOT\Applications` могут находиться и другие подразделы. Например, в ветви `HKEY_CLASSES_ROOT\Applications\explorer.exe` может располагаться подраздел `drives`. С его помощью можно переопределить файлы рисунков, применяемые в качестве значков логических дисков. Для этого достаточно в подразделе `drives` создать вложенный подраздел с названием соответствующим букве диска, значок которого нужно изменить, а в созданном подразделе нужно создать еще один — `DefaultIcon`. Параметр `(Ïî óìîë÷àíèþ)` этого подраздела как раз и определяет значок, используемый для отображения данного диска. Например, чтобы сменить значок диска `H:`, нужно изменить параметр `(Ïî óìîë÷àíèþ)` ветви реестра `HKEY_CLASSES_ROOT\Applications\explorer.exe\drives\h\DefaultIcon`.

- `MIME` — содержит сведения обо всех типах MIME, зарегистрированных на компьютере. Описание всех этих типов находится в ветви системного реестра `HKEY_CLASSES_ROOT\MIME\Database\Content Type`, включающей в себя подразделы с именами, соответствующими типу (`audio/basic`, `image/bmp`, `text/plain` и т. д.). Эти подразделы могут хранить следующие параметры:
  - `CLSID` — определяет CLSID-номер сервера ActiveX, который будет обслуживать данный тип;

  - `Extension` — устанавливает расширение файла, которое использует данный тип MIME;
  - `Encoding` — данный параметр `BINARY`-типа определяет код для кодировки данных MIME.

- `AppID` — предназначен для определения настроек удаленной активизации и защиты различных ActiveX-объектов, описанных в разделе `CLSID`. Как и `CLSID`, раздел `AppID` хранит список подразделов, названных в честь конкретного CLSID-номера ActiveX-объекта. О параметрах и подразделах этих ветвей реестра здесь рассказано не будет. Тем не менее если эта тема вам интересна, то в базе данных по реестру, поставляемой вместе с книгой, есть сведения о параметрах и подразделах этой ветви реестра.

Глава 5

Настройка оболочки

- **Значки**
- **Проводник**
- **Диалоги**
- **Другие возможности**

В предыдущей главе была полностью рассмотрена структура корневого раздела HKEY_CLASSES_ROOT. Структура остальных корневых разделов в этой книге рассматриваться не будет, так как она по своей природе не статична — нельзя точно предположить, какое название параметра решат использовать программисты при написании своего приложения и в какой ветви им захочется его создать. Поэтому структура таких корневых разделов системного реестра, как HKEY_CURRENT_USER и HKEY_LOCAL_MACHINE (за исключением ветви HKEY_LOCAL_MACHINE\SYSTEM), не будет рассматриваться вообще. Вместо этого будут описаны отдельные параметры, которые могут находиться в данных корневых разделах.

При этом стоит еще сказать, что целью создания в этой книге части о реестре было описание параметров реестра, доступ к которым нельзя получить никакими другими стандартными методами, кроме как с помощью реестра. Поэтому тем, кому интересна эта тема, еще раз советуем купить отдельную книгу, посвященную именно этому вопросу.

Значки

Но начнем наконец рассматривать параметры реестра. Для начала будут описаны различные настройки значков файлов или ActiveX-объектов, которые можно изменить в операционной системе Windows XP (эта глава не содержит информации о настройках из корневого раздела HKEY_CLASSES_ROOT).

Изображения

Все уже привыкли к стандартным значкам Windows, таким как значок диска, файла неизвестного расширения, папки и т. д. Но иногда все-таки хочется разнообразия, поэтому сейчас попробуем изменить некоторые стандартные значки, применяемые в операционной системе для файлов расширений. Для этого понадобится ветвь реестра HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\Shell Icons. Она как раз и содержит список параметров строкового типа, определяющих пути к файлам рисунков, которые будут использоваться при следующем входе в систему для построения изображений стандартных значков. Значения этих строковых параметров определяют путь к файлу изображения, а имя параметра соответствует индексу данного значка в библиотеке shell32.dll (этот индекс указывает на изображение по умолчанию для данного значка). Список всех индексов и соответствующих им значков можно просмотреть в приложении 5. Пока, чтобы не отвлекаться на перелистывание книги, скажем, что для отображения значка дисковода используется значок с индексом 6, для отображения «руки» под общедоступной папкой используется значок с индексом 28, а для отображения значка файлов справки используется значок с индексом 23. Попробуем изменить файлы рисунков для данных индексов.

По умолчанию ветвь HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\Shell Icons не присутствует в реестре, поэтому ее придется создать самостоятельно. После этого нужно создать в ней параметр строкового типа, имя которого равно 6 (для значка дисковода), а значение параметра

определяет путь к новому файлу рисунка. Аналогично создаются параметры для изменения других значков.

ПРИМЕЧАНИЕ

Не всегда внесенные изменения вступают в силу сразу после перезагрузки — иногда нужно подождать некоторое время, чтобы система внесла изменения в файл shelliconcache (данный файл используется как кэш, содержащий все файлы изображений, используемые для отображения значков в системе). Если же ждать не хочется, то можно попробовать самому удалить этот файл. Для этого нужно в командной строке ввести del shelliconcache или самостоятельно найти данный файл и удалить его (файл является суперскрытым, поэтому, чтобы он отобразился, необходимо установить соответствующий флажок на вкладке Вид диалога Свойства папки).

Кстати, существует также возможность изменения размера этого файла. Для этого предназначен параметр строкового типа Max Cached Icons, расположенный в ветви реестра HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer.

На рис. 5.1 можно увидеть результат изменения.

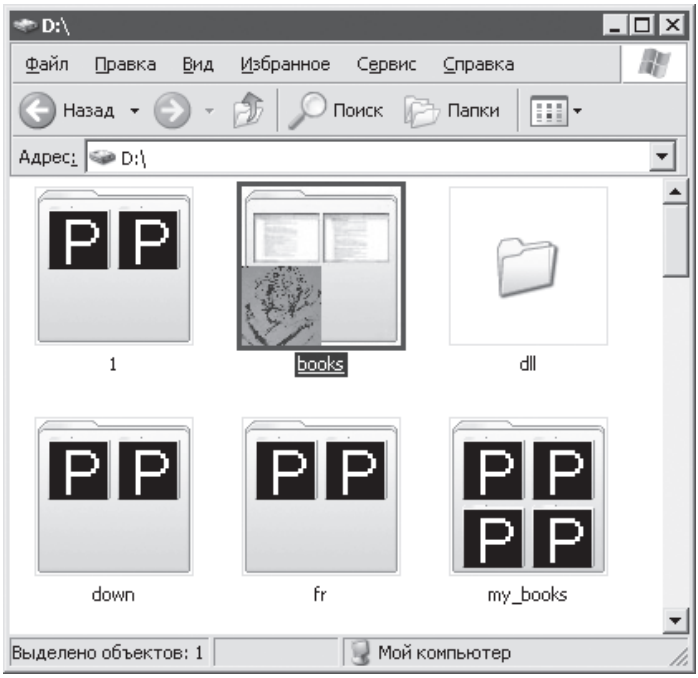


Рис. 5.1. Изменение изображения, используемого для отображения «руки» общедоступных папок и значка дисковода

Другой возможностью, которую предоставляет операционная система Windows, является изменение значков отдельно для каждого логического диска. Это дела-

ется с помощью ветви реестра HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\DriveIcons. Ветвь должна содержать разделы, каждый из которых назван в честь буквы логического диска (например, для диска C: создаваемый подраздел должен называться Ñ). Внутри раздела диска могут находиться еще два подраздела. Первый из них называется DefaultIcon — его параметр (Ïî óìîë÷àíèþ) как раз и определяет путь к значку, который будет использоваться для отображения дисков. Второй же подраздел называется DefaultLabel — если он присутствует, то значение его параметра (Ïî óìîë÷àíèþ) определяет название метки данного логического диска, применяемой, если диск не имеет своей собственной метки.

ПРИМЕЧАНИЕ

Не забывайте и о ветви реестра HKEY_CLASSES_ROOT\Applications\explorer.exe\drives\«буква диска»\DefaultIcon, с помощью которой также можно изменить значок для конкретного диска.

Кроме значка и метки диска, операционная система Windows XP позволяет выполнить еще один хакинг — определить, в каком месте названия будет находиться буква логического диска. По умолчанию буква отображается в конце названия, что в некоторых случаях не совсем удобно — буква может скрываться, если название слишком длинное. Поэтому, чтобы исправить этот недостаток, можно воспользоваться DWORD-параметром ShowDriveLettersFirst, расположенным в ветви реестра HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer. Он может принимать следующие значения:

- 1 — установлено по умолчанию, что как раз и соответствует отображению значка диска справа от его метки;
- 2 — вообще скрывает букву диска из его названия;
- 4 — буква диска будет отображаться слева от его метки, то есть название, которое раньше отображалось как, например, System (C:), после присвоения данному параметру значения 4 и последующей перезагрузки компьютера будет отображаться как (C:) System.

При выполнении хакинга со значками диска следует учитывать, что если вы в данный момент используете вид Эскизы страниц и описываемый в предыдущем разделе параметр thumbnail для идентификатора диска содержит путь к рисунку, то будет применяться именно рисунок из параметра thumbnail, а не изменяемый значок диска.

Следует также учитывать, что существует еще один параметр, определяющий изображение, которое будет накладываться поверх эскиза папки (если вы будете изменять эскиз для папки с помощью параметра thumbnail). Он находится в ветви реестра HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\Shell\Bags\AllFolders\Shell и называется Logo (имеет строковый тип). Его значение

определяет путь к рисунку, который будет накладываться поверх папки при использовании режима отображения Эскизы страниц.

Раз уж выше была затронута тема эскизов страниц, то мы еще немного поговорим и о них. Параметры отображения эскизов расположены в ветви системного реестра HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer. Эта ветвь может содержать два параметра DWORD-типа, имеющих названия ThumbnailQuality и ThumbnailSize. Первый определяет качество создаваемых системой эскизов страниц и может принимать значения от 50 до 100 (значение по умолчанию — 90). Второй же параметр определяет размер эскизов. Он может принимать значения от 32 до 255 (по умолчанию его значение равно 96).

Теперь несколько слов о настройке цвета названий различных значков операционной системы Windows. Здесь не будет рассказано о том, что можно изменить с помощью диалога Свойства: Экран, поэтому получится не очень много, но тем не менее.

Существует возможность изменения цвета зашифрованных и сжатых файлов в операционной системе Windows. Для этого применяются два параметра REG_BINARY-типа, имеющие следующий формат: 0õR 0õG 0xB 00, где 0õR определяет красную составляющую цвета, 0õG — зеленую, а 0xB — синюю (например, значение 00FF0000 определяет зеленый цвет имени файлов соответствующего типа). Эти параметры расположены в ветви реестра HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer. Первый из этих параметров называется AltColor и определяет цвет сжатых файлов, а второй — AltEncryptionColor — служит для изменения цвета зашифрованных файлов.

Последний трюк относится к цвету отображения названия определенного ActiveX-объекта. Все дело в том, что в Windows существует возможность указать операционной системе, что название конкретного ActiveX-объекта будет отображаться синим цветом (или цветом, который будет определен вышеописанным параметром AltEncryptionColor). Для этого используется параметр Attributes, расположенный в разделе ShellFolder ветви реестра, определяющей необходимый ActiveX-объект. В предыдущей главе уже рассматривался этот параметр, но еще не упоминалось о такой возможности.

Итак, чтобы отобразить название ActiveX-объекта, расположенное под его значком, синим цветом, достаточно установить данному параметру DWORD-типа битовую маску 0x04000000. Например, чтобы изменить цвет названия значка Панель задач и меню Пуск, расположенного в папке Панель управления, необходимо отредактировать параметр Attributes из ветви реестра HKEY_CLASSES_ROOT\clsid\{0DF44EAA-FF21-4412-828E-260A8728E7F1}\ShellFolder. К сожалению, на черно-белом изображении изменение цвета практически незаметно, поэтому рисунок не приводится.

Можно также заставить систему отобразить название ActiveX-объекта зеленым цветом (или цветом, который будет определен вышеописанным параметром AltColor). Для этого нужно присвоить параметру Attributes битовую маску 0õ00002000.

Следует только учесть, что в этом случае битовая маска 0x04000000 должна быть сброшена.

Последняя возможность битовой маски Attributes, которая будет рассмотрена, — эффект полупрозрачности значка ActiveX-объекта. Чтобы добиться такого эффекта, нужно присвоить параметру Attributes битовую маску 0õ00008000.

ПРИМЕЧАНИЕ

Данный эффект нельзя применить к значку папки Мой компьютер. Существует также одна особенность его применения для значка Корзины — при входе пользователя в систему полупрозрачность для него не применяется, но как только пользователь наведет на значок указатель мыши (и некоторое время подержит на нем), она станет полупрозрачной.

Расположение

Кроме изменения значков различных файлов Windows, существует возможность управления расположением значков на Рабочем столе. Само расположение значков описано в разделе HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\Streams\Desktop, поэтому если вы уже идеально настроили свой Рабочий стол, то можно запретить полный доступ к этому разделу системе, своей учетной записи и группе, к которой принадлежит ваша учетная запись, оставив только доступ на чтение.

Можно также заставить значок Корзины при следующем входе в систему отображаться в правом нижнем углу. Для этого нужно присвоить DWORD-параметру AdjustRecycleBinPosition значение, равное 1 (рис. 5.2). Он расположен в ветви реестра HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\ScreenResFixer. После следующего входа в систему Windows, независимо от настроек раздела HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\Streams\Desktop, отобразит значок Корзины в правом нижнем углу. После этого система изменит значение параметра AdjustRecycleBinPosition на 2 (то есть разрешит изменение расположения значка Корзины). Если же вы не хотите, чтобы расположение этого значка можно было изменить, то следует также запретить полный доступ и к этой ветви реестра, оставив только доступ на чтение.

ПРИМЕЧАНИЕ

Есть возможность определить порядок расположения ActiveX-объектов на Рабочем столе. Для этого применяется DWORD-параметр SortOrderIndex, расположенный в разделе ActiveX-объекта ветви реестра HKEY_CLASSES_ROOT\clsid. Например, если значение этого параметра для значка Корзины будет равно 0x00000060, для значка папки Мой компьютер — 0x00000054, а для значка Мои документы — 0x00000048, то на первом месте Рабочего стола будет отображаться Корзина, на втором — Мой компьютер, а на третьем — Мои документы.

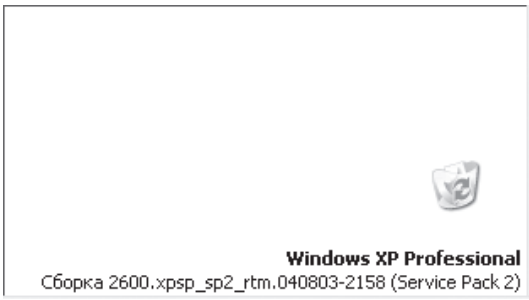


Рис. 5.2. Расположение значка Корзины

Отображение

Еще одной возможностью, которую предоставляет пользователям Windows, является определение тех значков ActiveX-объектов, которые будут отображаться на Рабочем столе, в папках Панель управления и Мой компьютер.

ActiveX-объекты

Для определения тех значков ActiveX-объектов, которые будут отображаться на Рабочем столе, в папках Панель управления и Мой компьютер, применяются следующие ветви реестра.

- `HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\Desktop\NameSpace` — определяет те из ActiveX-объектов, которые будут отображаться на Рабочем столе. Раздел `NameSpace` содержит набор дочерних подразделов, названных в честь CLSID-номера ActiveX-объекта, который должен отображаться на Рабочем столе. Например, чтобы отобразить на Рабочем столе значок, вызывающий диалог Запуск программы, необходимо создать в данном разделе подраздел `{2559a1f3-21d7-11d4-bdaf-00c04f60b9f0}`. Результат можно видеть на рис. 5.3.

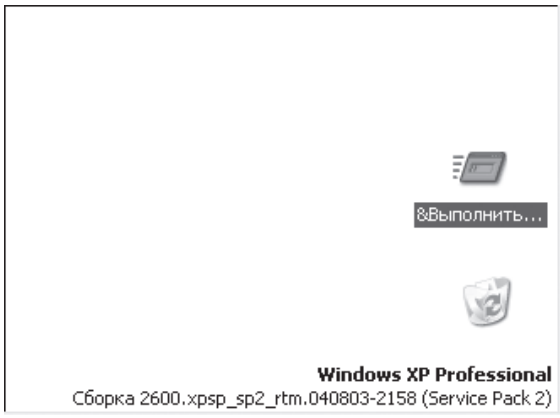


Рис. 5.3. Отображение на Рабочем столе значка Запуск программы

Раздел NameSpace можно создать в аналогичной ветви корневого раздела HKEY_LOCAL_MACHINE. В этом случае ActiveX-объект будет отображаться на Рабочем столе всех пользователей компьютера, а не только данного пользователя.

ВНИМАНИЕ

Данная ветвь реестра, кроме значков ActiveX-объектов, которые будут отображаться на Рабочем столе, включает в себя еще и определение CLSID-номера {1f4de370-d627-11d1-ba4f-00a0c91eedba}. Не удаляйте определение этого номера, так как оно необходимо для корректной работы с функцией поиска и его удаление сделает невозможным использование диалога, вызываемого комбинацией клавиш Windows+F.

- HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\HideDesktopIcons — если предыдущая ветвь реестра определяла те из значков ActiveX-объектов, которые будут отображаться на Рабочем столе, то эта ветвь устанавливает совершенно противоположную возможность — она указывает на те ActiveX-объекты, которые не будут отображаться на Рабочем столе. При этом можно указать отдельные наборы скрываемых значков в зависимости от типа меню Пуск, используемого в данный момент, ведь данная ветвь хранит два дочерних раздела — ClassicStartMenu и NewStartPanel. Если вы определите CLSID-номера в разделе ClassicStartMenu, то данные значки ActiveX-объектов не будут отображаться на Рабочем столе при использовании классического меню Пуск. Если же вы определите CLSID-номера в разделе NewStartPanel, то данные значки ActiveX-объектов не будут отображаться на Рабочем столе при использовании нового меню Пуск.

  Чтобы определить CLSID-номер скрываемого ActiveX-объекта, достаточно в необходимом разделе создать параметр DWORD-типа, названный в честь этого CLSID-номера. Значение параметра должно быть равно 1 (если значение равно 0, то значок ActiveX-объекта будет отображаться). Например, чтобы скрыть значок диалога Запуск программы, который был недавно добавлен, необходимо создать в ветви реестра HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\HideDesktopIcons\ClassicStartMenu (если используется классическое меню Пуск) параметр DWORD-типа {2559a1f3-21d7-11d4-bdaf-00c04f60b9f0}, значение которого должно быть равно 1.

  Данную ветвь можно использовать и в корневом разделе HKEY_LOCAL_MACHINE.

ПРИМЕЧАНИЕ

Существует один интересный CLSID-номер ActiveX-объекта, описания которого вы не встретите в ветви реестра HKEY_CLASSES_ROOT\CLSID. Это CLSID-номер {00000000-0000-0000-0000-000000000000}. Если запретить отображение данного CLSID-номера на Рабочем столе с помощью вышеописанной ветви реестра, то после перезагрузки оболочки на Рабочем столе останутся только значки Корзина и Мой компьютер — остальные ActiveX-объекты, папки и файлы Рабочего стола будут скрыты.

- HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\MyComputer\NameSpace — определяет список CLSID-номеров ActiveX-объектов, которые будут отображаться в папке Мой компьютер. Формат данной ветви полностью аналогичен формату описанной выше ветви реестра HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\ Desktop\NameSpace. Если использовать такую ветвь реестра, расположенную в корневом разделе HKEY_LOCAL_MACHINE, то определяемый ею значок будет отображен для всех пользователей системы.

ПРИМЕЧАНИЕ

Но между приведенными ветвями реестра для различных корневых разделов существует небольшое отличие — ветвь реестра из корневого раздела HKEY_LOCAL_MACHINE также может хранить раздел DelegateFolders, содержимым которого является подраздел {59031a47-3f72-44a7-89c5-5595fe6b30ee}. Этот подраздел определяет общие папки, отображаемые в папке Мой компьютер, и если удалить или просто переименовать раздел DelegateFolders, то общие папки в папке Мой компьютер отображаться не будут.

Например, можно попробовать самостоятельно добавить в папку Мой компьютер значок ActiveX-объекта. На этот раз нужно создать свой собственный ActiveX-объект.

Как уже говорилось ранее, для создания уникального CLSID-номера ActiveX-объекта применяется программа guidgen.exe, входящая в стандартную поставку таких компиляторов, как Microsoft Visual Studio .NET, а также Microsoft Visual C++. После запуска данной программы перед вами появится диалог, в котором требуется определить, какой именно уникальный идентификатор нужно создать. Чтобы создать CLSID-номер, необходимо установить переключатель 4. Registry Format. После выбора типа уникального идентификатора нужно выбрать сам созданный идентификатор — нажимайте кнопку Next GUID до тех пор, пока программа не создаст идентификатор, который вам понравится. После этого достаточно нажать кнопку Copy, чтобы поместить этот идентификатор в буфер обмена.

Все, уникальный CLSID-номер у вас уже есть. Например, таким номером является {23D0F57C-5E2C-4fb2-BE50-B27DBD7EFB76}, созданный с помощью программы guidgen.exe. После создания CLSID-номера нужно зарегистрировать его в системе. Для этого используется ветвь системного реестра HKEY_CLASSES_ROOT\CLSID, формат содержимого которой был описан в предыдущей главе. Создайте в этой ветви дочерний раздел, названный в честь CLSID-номера, созданного программой guidgen. Теперь необходимо назвать значок ActiveX-объекта — для этого используется параметр (Ïî óìîë÷àíèþ) созданного раздела. После этого нужно зарегистрировать для созданного ActiveX-объекта значок, с этой целью применяется параметр DefaultIcon дочернего раздела ветви созданного CLSID-номера. После создания значка нужно скрыть все стандартные команды его контекстного меню — для этого при-

свойте DWORD-параметру Attributes значение 0. Он должен находиться в разделе ShellFolder ветви созданного CLSID-номера. И наконец, можно создавать содержимое пользовательского контекстного меню — для этого используется раздел shell ветви CLSID-номера. Создайте в нем необходимые глаголы, а в них — подраздел command, параметр (Ïî óìîë÷àíèþ) которого и будет определять команду, вызываемую при выборе из контекстного меню вашего значка соответствующего действия.

Теперь у вас есть не только уникальный CLSID-номер, но и свой собственный ActiveX-объект. Осталось только создать подраздел {23D0F57C-5E2C-4fb2-BE50-B27DBD7EFB76} в ветви реестра HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\MyComputer\NameSpace и смотреть на то, что получилось (рис. 5.4).

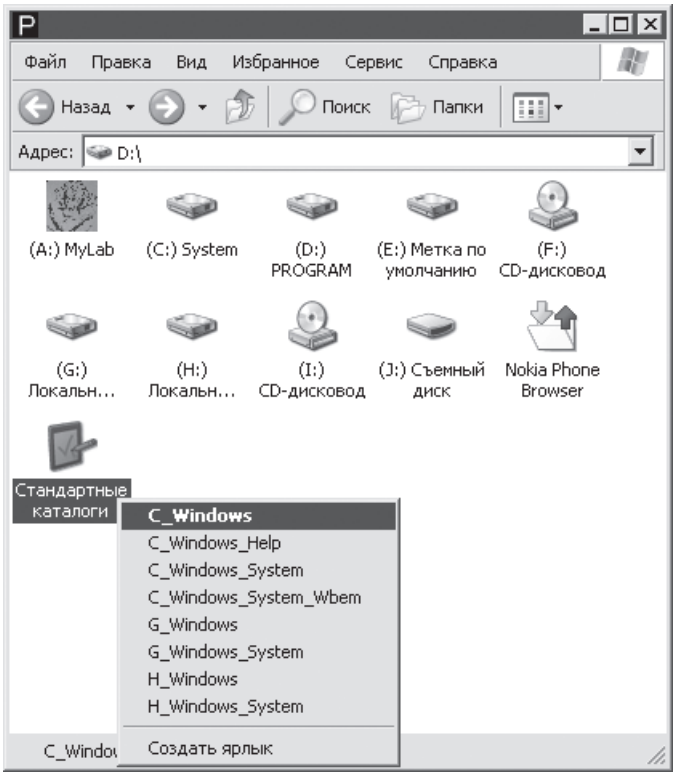


Рис. 5.4. Создание своего ActiveX-объекта и расположение его в папке Мой компьютер

- HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\HideMyComputerIcons — скрывает значки из папки Мой компьютер. Ее формат аналогичен формату уже описанной ветви системного реестра HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\HideDesktopIcons, поэтому рассматривать ее мы не будем. Эту ветвь можно использовать и в корневом разделе HKEY_LOCAL_MACHINE.

Например, чтобы скрыть только что созданный значок ActiveX-объекта из папки Мой компьютер, необходимо создать DWORD-параметр {23D0F57C-5E2C-4fb2-BE50-B27DBD7EFB76} и присвоить ему значение, равное 1. Создавать его нужно в ветви реестра HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\HideMyComputerIcons.

- HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\ControlPanel\NameSpace — содержит ссылки на те ActiveX-объекты, значки которых будут отображаться в папке Панель управления. Его формат аналогичен формату описанных выше разделов реестра, предназначенных для добавления значков ActiveX-объектов.
- HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\NetworkNeighborhood\NameSpace — определяет те ActiveX-объекты, значки которых будут отображаться в папке Сетевое окружение. Его формат аналогичен формату описанных выше разделов реестра, предназначенных для добавления значков ActiveX-объектов.
- HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\RemoteComputer\NameSpace — определяет те ActiveX-объекты, значки которых будут отображаться в папке Удаленные компьютеры. Его формат аналогичен формату описанных выше разделов реестра, предназначенных для добавления значков ActiveX-объектов.

  По умолчанию в этой ветви реестра присутствуют только ссылки на ActiveX-объекты Принтеры и Назначенные задания. Если вы как администратор не хотите, чтобы эти ActiveX-объекты отображались в папке удаленного компьютера, то можно их удалить.
- HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\policies\NonEnum — раздел более категоричен, чем все рассмотренные ранее, — он определяет те из ActiveX-объектов, которые вообще не будут отображаться в оболочке пользователей (или конкретного пользователя, если применяется ветвь из корневого раздела HKEY_CURRENT_USER). Все ActiveX-объекты, описанные в этой ветви, не будут применяться при построении пользовательского интерфейса.

  Чтобы запретить использование какого-либо ActiveX-объекта, достаточно создать в данной ветви реестра параметр DWORD-типа, названный в честь CLSID-номера скрываемого ActiveX-объекта, и присвоить ему значение, равное 1.

Хотелось бы сказать еще об одной недокументированной возможности скрытия ActiveX-объекта из всех мест оболочки, где он используется, — о параметре Attributes, находящемся в подразделе ShellFolder раздела скрываемого нами ActiveX-объекта. Содержимое данного параметра уже было описано в предыдущей главе, но не упоминалось об этой возможности.

Итак, чтобы скрыть значок ActiveX-объекта, достаточно его параметру Attributes присвоить битовую маску 0õ00100000.

Скрытие CPL-файлов

Кроме скрытия ActiveX-объектов, существует еще и возможность скрытия CPL-файлов из содержимого окна Панель управления. Для этого могут применяться две стандартные ветви реестра Windows, первой из которых является ветвь реестра HKEY_CURRENT_USER\Control Panel\don't load. Она может хранить список параметров строкового типа, названных в честь CPL-файлов, значения которых не важны. Все CPL-файлы, чьи имена описаны в качестве параметров ветви HKEY_CURRENT_USER\Control Panel\don't load, не будут отображаться в Панели управления. Например, чтобы скрыть апплет Мышь, необходимо в данной ветви реестра создать параметр строкового типа main.cpl (будет также скрыт апплет Клавиатура).

Предыдущая ветвь имеет один существенный недостаток применения — пользователь может удалить все созданные вами параметры. Если же вы не хотите, чтобы он мог это сделать, то нужно запретить полный доступ к данной ветви реестра, оставив только доступ на чтение. Можно просто воспользоваться другой ветвью реестра — HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Control Panel\don't load. Она имеет такой же формат, что и предыдущая ветвь, но скрывает апплеты Панели управления для всех пользователей системы.

Другие возможности

Теперь рассмотрим другие возможности реестра, которые можно использовать в своей системе.

- Чтобы при создании ярлыка файла к его названию не добавлялся префикс Ярлык для, необходимо присвоить параметру REG_BINARY-типа link значение 0. Он расположен в ветви реестра HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer (можно также использовать корневой раздел HKEY_LOCAL_MACHINE).
- Еще одной возможностью, которую предоставляет пользователю Windows, является определение области вокруг значка, в пределе которой его перемещение не будет считаться перемещением как таковым — значок вернется на свое прежнее место (довольно часто при быстром двойном щелчке на значке можно его случайно переместить, что может испортить весь стиль оформления Рабочего стола). Для этого используются два параметра строкового типа, расположенных в ветви реестра HKEY_CURRENT_USER\Control Panel\Desktop. Первый из этих параметров называется DragHeight и определяет расстояние в пикселах, в пределах которого перемещение значка или окна по вертикали не приводит к перетаскиванию. Например, если значение этого параметра равно 35, то при перемещении значка на 35 пикселов вверх/вниз он останется на своем прежнем месте. Второй параметр называется DragWidth — определяет расстояние в пикселах, в пределах которого перемещение значка или окна по горизонтали не приводит к перетаскиванию.

Проводник

Стандартные окна операционной системы являются еще одним элементом оболочки Windows, нестандартные настройки которого могут быть интересны. Окна имеют не много параметров настройки, доступ к которым нельзя получить из пользовательских диалогов, но все-таки эти параметры стоят того, чтобы о них написать.

Оформление

В стандартной поставке операционная система Windows имеет довольно строгий стиль оформления. Это можно заметить уже при первом взгляде на панель инструментов и меню окон Проводника. Уже стало стандартом то, как данные элементы окна выглядят, и кажется, что любой другой стиль оформления данных элементов будет неправильным и излишним. Но так ли это? Попробуйте изменить элементы оформления окна, а через некоторое время решите, оставить ли эти изменения в системе. Для себя автор данной книги уже решил — обязательно оставить, так как с некоторых пор стандартный стиль оформления окон ему кажется просто ужасным.

Итак, что же можно изменить? Всем известен трюк с изменением фона панели инструментов. Для его реализации понадобится один параметр строкового типа и одна ветвь реестра. Возьмем ветвь реестра `HKEY_CURRENT_USER\Software\Microsoft\Internet Explorer\Toolbar`. Чтобы изменить рисунок фона панели инструментов для Проводника, достаточно создать в этой ветви строковый параметр `BackBitmapShell` и присвоить ему путь к файлу рисунка, который будет использоваться для отображения в качестве фона.

Теперь, если стандартный логотип Microsoft не подходит по цвету или стилевому оформлению к только что измененной панели инструментов, можно изменить и его. Для этого используется та же ветвь реестра, что и при изменении фона панели инструментов, — `HKEY_CURRENT_USER\Software\Microsoft\Internet Explorer\Toolbar`. Но теперь вам понадобятся сразу два строковых параметра — один для указания пути к файлу рисунка, используемого для отображения логотипа Microsoft при обычном виде Проводника, а второй — для указания пути к файлу рисунка, используемого при полноэкранном виде Проводника (после нажатия клавиши F11). Первым из этих параметров является параметр строкового типа `SHBigBitMap`. Вторым — параметр строкового типа `SHSmallBitMap`.

ВНИМАНИЕ

Кроме того, нужно изменить значения параметров строкового типа BrandBitMap, а также SmBrandBitMap, которые расположены в той же ветви реестра, что и описываемые параметры. Пока что просто присвойте им те же значения, что и параметрам SHSmallBitMap и SHBigBitMap — о них будет рассказано позже. Если не изменить значения параметров BrandBitMap и SmBrandBitMap, то изменить логотип Проводника не получится.

На рис. 5.5 можно увидеть, что получилось.

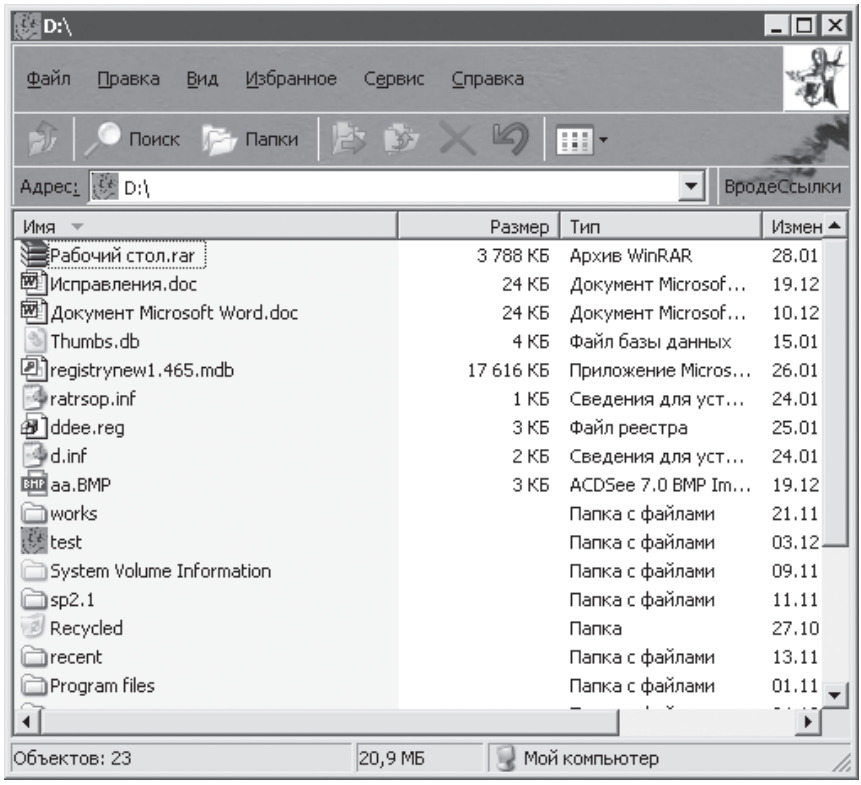


Рис. 5.5. Изменение логотипа и фона панели инструментов

ПРИМЕЧАНИЕ

Еще одной возможностью, которую позволяет выполнить оболочка, является уменьшение значка логотипа. Для этого применяется DWORD-параметр BrandHeight, расположенный в указанной ветви реестра. Если его значение равно 50 или меньше, то будет использоваться стандартный размер логотипа, а если значение от 60 до 800, то уменьшенный.

Операционная система Windows позволяет выполнить еще один хакинг — увеличение высоты панели инструментов. Для этого опять-таки применяется ветвь реестра `HKEY_CURRENT_USER\Software\Microsoft\Internet Explorer\Toolbar`. Чтобы увеличить высоту панели инструментов, достаточно создать в этой ветви реестра DWORD-параметр `SmBrandHeight` и присвоить ему значение, равное количеству пикселов, на которое вы хотели бы увеличить панель инструментов. Например, на рис. 5.6 можно увидеть то же изображение, что и на рис. 5.5, но с использованием параметра `SmBrandHeight`, значение которого равно `50`.

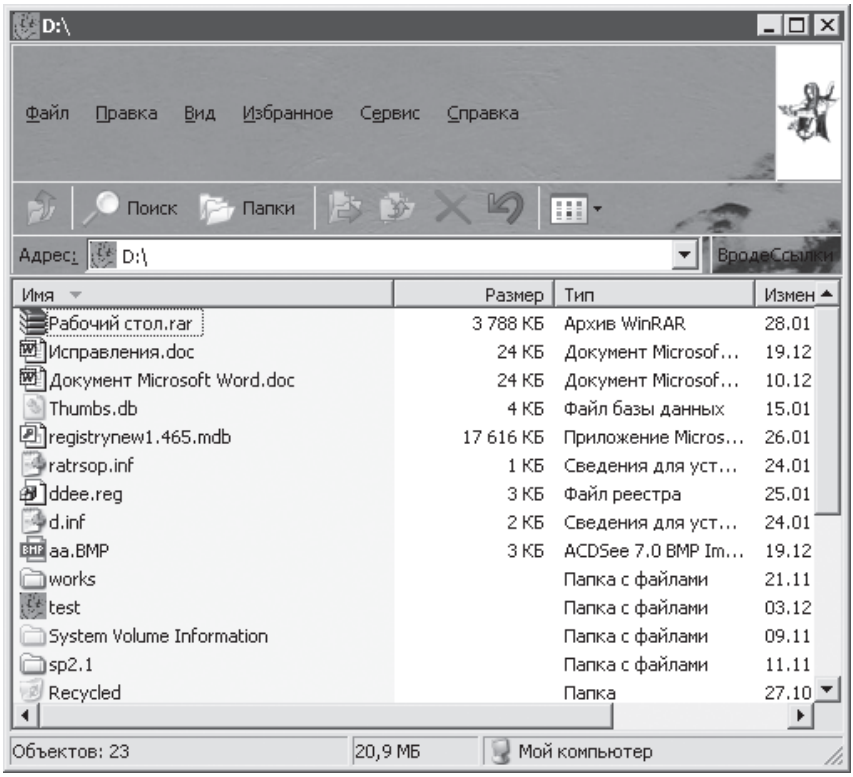


Рис. 5.6. Увеличение размера панели инструментов

Стандартные папки Windows

Еще одной возможностью, которая вам может быть интересна, является изменение путей к стандартным папкам Windows, таким как Мои документы, Избранное, Моя музыка, Program Files и т. д. Для реализации этой возможности используется несколько ветвей реестра, но большая часть параметров, определяющих пути к папкам, без сомнения, находится в ветви реестра `HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\Shell Folders`. Вот именно с содержимого данной ветви нужно начинать (все параметры этой ветви имеют тип `REG_SZ`).

ПРИМЕЧАНИЕ

В реестре Windows существует ветвь HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\Shell Folders, определяющая путь к папкам, применяемым всеми пользователями, а не конкретным.

- `Administrative Tools` — определяет путь к папке Администрирование, содержимое которой используется для построения одноименной ветви меню Пуск.

- AltStartup — хранит путь к папке, содержимое которой будет запускаться при входе пользователя в систему. Разница между данным параметром реестра и параметром Startup состоит в том, что ссылки из каталога, определяемого параметром AltStartup, не отображаются на вкладке Автозагрузка программы msconfig.exe.
- AppData — определяет расположение каталога Application Data, который используется различными приложениями для хранения своих данных.
- Cache — указывает путь к папке Temporary Internet Files, используемой браузером Internet Explorer для хранения частей загружаемых документов.
- CD Burning — определяет путь к папке, используемой стандартной программой записи компакт-дисков для хранения самих компакт-дисков и необходимой информации (каталог CD Burning, содержащийся в подкаталоге Microsoft каталога Application Data).
- Cookies — хранит путь к папке Cookies, используемой браузером Internet Explorer.
- Desktop — определяет путь к папке Рабочий стол, содержащей все файлы, которые расположены на Рабочем столе текущего пользователя.
- Favorites — указывает путь к папке Избранное, используемой Internet Explorer для хранения ссылок на избранные страницы пользователя.
- Fonts — определяет путь к папке Шрифты, содержащей все шрифты, установленные на компьютере. Доступ к данной папке можно также получить с помощью ActiveX-объекта {D20EA4E1-3957-11d2-A40B-0C5020524152}.
- History — хранит путь к папке History, содержимое которой используется для формирования журнала всех сайтов, которые посещались пользователем в течение последних двух недель.
- Local AppData — содержит тот же путь, что и параметр AppData.
- Local Settings — указывает путь к папке Local Settings, содержимым которой являются папки, определяемые параметрами AppData, History, Cache.
- My Music — определяет путь к папке, которая будет использоваться такими программами, как, например, Проигрыватель Windows Media, для копирования в них музыкальных файлов.
- My Pictures — хранит путь к папке Мои рисунки.
- My Video — определяет путь к папке Мое видео.
- NetHood — указывает путь к папке NetHood, содержащей ссылки на сетевые папки, к которым вы получали доступ в последнее время.
- PrintHood — определяет путь к папке PrintHood, содержащей ссылки на сетевые принтеры, к которым вы получали доступ в последнее время.
- Personal — хранит путь к папке Мои документы. Конечно, местоположение папки Мои документы можно определить и с помощью диалога Свойства, но посредством данного параметра можно, например, указать логический диск в качестве папки Мои документы.

- `Programs` — определяет путь к папке Программы, содержимое которой используется для построения ветви Программы меню Пуск. Данную папку можно также вызвать с помощью ActiveX-объекта `{7be9d83c-a729-4d97-b5a7-1b7313c39e0a}`. При этом вы сможете просмотреть не только программы меню Пуск, отображаемые для вашей учетной записи, но и программы, отображаемые для всех учетных записей.
- `Recent` — хранит путь к папке Recent, содержащей ссылки на все файлы и папки, к которым вы получали доступ в последнее время.
- `SendTo` — определяет путь к папке SendTo, содержимое которой используется для построения списка Отправить контекстного меню файлов и папок.
- `Start Menu` — указывает путь к папке Главное меню, содержимое которой отображается в виде разделов меню Пуск.
- `Startup` — хранит путь к папке Автозагрузка, содержимое которой используется для построения соответствующей ветви меню Пуск. В отличие от папки, определяемой параметром `AltStartup`, содержимое этой папки можно просмотреть с помощью вкладки Автозагрузка программы `msconfig.exe`.
- `Templates` — определяет путь к папке Templates. Содержимое именно этой папки определяет те файлы шаблонов, которые будут создаваться после выбора соответствующей команды из списка Создать диалога контекстного меню Рабочего стола или Проводника, если для создания файла используется параметр реестра `FileName` (он содержится в корневом разделе `HKEY_CLASSES_ROOT` и описывался в предыдущей главе).

ПРИМЕЧАНИЕ

Если вы хотите удалить какой-нибудь параметр из приведенной выше ветви реестра, то необходимо будет удалить его также из ветви реестра HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\User Shell Folders, иначе через некоторое время он будет создан заново. Следует также учитывать, что лучше не удалять полностью параметры приведенной ветви реестра — лучше просто удалить их значения.

Другой ветвью, которая содержит некоторые пути к стандартным папкам Windows, является ветвь реестра `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion`. Для описания пути к таким папкам используются следующие параметры.

- `WallPaperDir` — определяет путь к папке, содержимое которой используется при построении списка Фоновый рисунок вкладки Рабочий стол диалога Свойства: Экран. Он является параметром `REG_EXPAND_SZ`-типа.
- `SM_GamesName` — этот параметр строкового типа указывает название папки, содержимое которой будет использоваться при построении соответствующего списка меню Пуск (по умолчанию списка Игры). В дальнейшем именно к этой папке будут обращаться программы при попытке записи в список Игры новых элементов.

- SM_AccessoriesName — параметр строкового типа, определяет название папки, содержимое которой будет использоваться при построении соответствующего списка меню Пуск (по умолчанию списка Стандартные). В дальнейшем именно к этой папке будут обращаться программы при попытке записи в список Стандартные новых элементов.

Некоторые пути к стандартным каталогам можно встретить в ветви реестра Windows HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Setup.

- DriverCachePath — определяет путь к папке Driver Cache, используемой для хранения резервных копий всех системных библиотек и программ операционной системы Windows. Если вы пользуетесь несколькими операционными системами Windows XP, то можно в каждой из них указать путь к общей папке, в которой будут храниться все резервные копии библиотек, используемых всеми операционными системами Windows XP.
- SourcePath — хранит путь к папке, содержащей установочные файлы текущей операционной системы. По умолчанию он определяет букву дисковода, с которого вы устанавливали операционную систему, и именно к этому дисководу система обращается, когда ей необходимо установить дополнительные компоненты. Если дистрибутив вашей операционной системы хранится у вас на жестком диске, то в этом параметре можно описать путь к нему (в этом случае нужно будет присвоить DWORD-параметру CDInstall данной ветви реестра значение, равное 0). Теперь при необходимости установки дополнительных компонентов Windows будет всегда обращаться к папке дистрибутива на вашем жестком диске, не мучая вас просьбой вставить диск дистрибутива в привод компакт-дисков.
- ServicePackSourcePath — определяет путь к папке, которая содержит файлы установленного на вашем компьютере пакета обновлений. Именно к этой папке Windows обращается, если ей необходимо заново установить файлы пакета обновлений (например, если оригинальные версии файлов изменены или удалены).

Конфигурация

Теперь рассмотрим несколько параметров строкового типа, предназначенных для настройки конфигурации Проводника.

- MenuShowDelay — определяет задержку перед отображением меню, которую оболочка будет ожидать. Он расположен в ветви реестра HKEY_CURRENT_USER\Control Panel\Desktop. По умолчанию его значение равно 400.
- BrowseNewProcess — определяет, будет ли открываться каждое окно Проводника и браузера Internet Explorer в виде отдельного процесса или все они будут открываться как часть одного процесса. Если значение этого параметра равно YES, то каждое окно Проводника и браузера Internet Explorer будет открываться как отдельный процесс. Плюсом этого метода можно считать большую стабильность — если возникнет ошибка в работе одного окна Проводника, то оно закроется, но остальные окна останутся рабочими. Если же значение равно NO,

то все окна Проводника и браузера Internet Explorer будут открываться как часть оболочки. Этот метод имеет как минусы, так и плюсы. К минусам можно отнести невысокую стабильность работы системы по сравнению с предыдущим методом — если возникнет ошибка в одном из окон Проводника или браузера, то будут закрыты все открытые окна и начнется перезагрузка самой оболочки Windows. К плюсам же можно причислить меньший размер занимаемой оперативной памяти при открытии нескольких окон, а также более быстрое открытие каждого нового окна Проводника или браузера.

Параметр расположен в ветви реестра `HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\BrowseNewProcess`.

Диалоги

Еще одной возможностью, которую предоставляет пользователям Windows, является редактирование различных списков стандартных диалогов операционной системы, а также скрытие некоторых вкладок диалогов без использования групповых политик и консоли управления безопасностью. Например, операционная система позволяет редактировать содержимое списка Дополнительные параметры на вкладке Вид диалога Свойства папки или содержимое списка Параметры на вкладке Дополнительно диалога Свойства: Интернет.

Содержимое списков

Для формирования содержимого некоторых списков стандартных диалогов Windows используется ряд стандартных параметров (о ветвях, используемых для формирования списков стандартных диалогов, будет рассказано чуть позже). Эти параметры создаются в дочерних папках ветвей (одна папка — один элемент списка), формирующих конкретные списки диалогов.

- `Text` — этот параметр строкового типа определяет текст надписи, отображаемой в соответствующем списке диалога. Надпись находится напротив подкаталога или флажка, который она идентифицирует.
- `HKeyRoot` — параметр `DWORD`-типа, указывает корневой раздел реестра, в котором находится параметр, изменяющийся при смене состояния соответствующего флажка или переключателя. Параметр может принимать следующие значения:
  - `0õ80000000` — находится в корневом разделе `HKEY_CLASSES_ROOT`;
  - `0õ80000001` — хранится в корневом разделе `HKEY_CURRENT_USER`;
  - `0õ80000002` — находится в корневом разделе `HKEY_LOCAL_MACHINE`;
  - `0õ80000003` — хранится в корневом разделе `HKEY_USERS`.
- `ValueName` — этот параметр строкового типа определяет название параметра, который будет меняться при изменении состояния соответствующего флажка или переключателя.
- `RegPath` — указывает остальной путь к ветви реестра, в которой будет находиться параметр, изменяющийся при смене состояния соответствующего

флажка или переключателя. Как видите, этим и двумя предыдущими параметрами можно задать полный путь к параметру, значение которого будет изменяться соответствующим элементом списка. Параметр имеет строковый тип.

- `DefaultValue` — определяет значение по умолчанию изменяемого описываемым элементом списка параметра. Это значение используется в том случае, когда в указанной ветви реестра не существует параметр, значение которого изменяется данным элементом списка. Тип параметра `DefaultValue` зависит от типа изменяемого данным элементом списка параметра. Например, если изменяемый параметра имеет тип `REG_SZ`, то и параметр `DefaultValue` будет иметь тип `REG_SZ`.
- `CheckedValue` — хранит значение, которое будет присвоено изменяемому параметру после установки соответствующего флажка или переключателя и нажатия кнопки Применить. Тип параметра `CheckedValue`, так же как и в предыдущем случае, зависит от типа параметра, изменяемого данным элементом списка.
- `UncheckedValue` — если предыдущий параметр определял значение изменяемого параметра при установке флажка или переключателя, то `UncheckedValue` определяет значение изменяемого параметра при снятом флажке (естественно, что при создании переключателя этот параметр не используется, так как переключатель не может быть снятым). Тип параметра `UncheckedValue`, так же как и тип двух предыдущих параметров, зависит от типа параметра, изменяемого данным элементом списка.
- `Bitmap` — этот параметр строкового типа определяет путь к файлу рисунка, используемого для отображения подпапки в списке. К сожалению, нет никакой возможности изменить рисунок флажка или переключателя с помощью данного параметра.
- `Mask` — если в разделе, определяющем соответствующий элемент списка, присутствует этот параметр `DWORD`-типа, то изменяемый параметр является битовой маской и система должна не переписать параметр заново, а просто изменить один из его битов — именно битовая маска этого бита и указывается в данном параметре.
- `Type` — этот параметр строкового типа определяет, какой именно элемент списка будет создан. Если значение этого параметра равно `group`, то будет создана подпапка. Если значение равно `checkbox`, то будет создан флажок, а если значение равно `radio`, то в списке появится переключатель.

Вот и все параметры, используемые системой для формирования одного элемента списка некоторых стандартных диалогов. Теперь перечислим все те ветви реестра, в которых могут находиться приведенные выше параметры, а также те диалоги, спискам которых они соответствуют.

- `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\Advanced` — определяет содержимое списка Дополнительные параметры вкладки Вид диалога Свойства папки.
- `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\VisualEffects` — хранит содержимое списка Особые эффекты,

отображаемого на вкладке Визуальные эффекты диалога Параметры быстродействия. Чтобы отобразить этот диалог, необходимо нажать кнопку Параметры, расположенную на вкладке Дополнительно диалога Свойства системы.

- HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Internet Explorer\AdvancedOptions — определяет содержимое списка Параметры вкладки Дополнительно диалога Свойства: Интернет.
- HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\StartMenu — хранит содержимое списка Дополнительные параметры меню "Пуск" диалога Настройки классического меню "Пуск". Чтобы вызвать данный диалог, необходимо нажать кнопку Настроить, расположенную напротив переключателя Классическое меню "Пуск" на вкладке Меню "Пуск" диалога Свойства панели задач и меню "Пуск".
- HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\StartMenu\StartPanel — определяет содержимое списка Элементы меню "Пуск", расположенного на вкладке Дополнительно диалога Настройка меню "Пуск". Чтобы вызвать данный диалог, необходимо нажать кнопку Настроить, расположенную напротив переключателя Меню "Пуск" на вкладке Меню "Пуск" диалога Свойства панели задач и меню "Пуск".
- HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\SO — хранит содержимое списка Параметры диалога Параметры безопасности. Этот диалог можно вызвать с помощью нажатия кнопки Другой на вкладке Безопасность диалога Свойства: Интернет браузера Internet Explorer. Аналогичные настройки можно встретить в ветви системного реестра HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\SOIEAK.

Например, можно добавить свой элемент списка в один из диалогов, чтобы больше не искать его в реестре при необходимости модификации (рис. 5.7).

Скрытие вкладок и других элементов диалогового окна

Теперь поговорим о параметрах реестра Windows, предназначенных для скрытия различных вкладок стандартных диалогов. В данном разделе книги не будут упоминаться параметры, используемые консолью mmc.exe, — этому посвящена отдельная глава книги. Сейчас же будут рассмотрены параметры, используемые самой операционной системой, хотя их не так уж и много.

- HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\DateTime — содержит сведения о серверах, используемых для синхронизации системных часов с указанным сервером времени. Но, кроме этого, она может хранить DWORD-параметр Support Internet Time. Если его значение равно 0, то из диалога Дата и время исчезнет вкладка Время Интернета, которая применяется для настройки синхронизации системного времени с сервером Интернета.

Рис. 5.7. Добавление своих элементов списка диалогового окна

- `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Shared Tools\MSConfig` — определяет настройки стандартной программы `msconfig`. Одним из параметров этой ветви может быть `DWORD`-параметр `boot.ini`. Если он действительно содержится в приведенной ветви реестра, то из программы `msconfig.exe` исчезнет вкладка `BOOT.INI`, предназначенная для облегчения создания данного файла в мультизагрузочных системах.
- `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall\«íàçâàíèå ïðîãðàììû»` — хранит всю информацию, необходимую для построения элемента списка апплета `Установка и удаление программ`, определяемого разделом «íàçâàíèå ïðîãðàììû». Но здесь не будут перечислены те параметры реестра, которые могут находиться в приведенной выше ветви. Намного интереснее рассмотреть несколько параметров данной ветви,

с помощью которых можно заблокировать те или иные кнопки для соответствующего элемента списка.

- `NoRemove` — параметр `DWORD`-типа определяет, будет ли для соответствующей программы в списке апплета Установка и удаление программ отображаться кнопка Удалить. Если значение данного параметра равно `1`, то для элемента списка, определяющего программу, кнопка Удалить отображаться не будет.
- `NoModify` — этот параметр `DWORD`-типа определяет, будет ли для соответствующей программы в списке апплета Установка и удаление программ отображаться кнопка Изменить. Если значение данного параметра равно `1`, то для элемента списка, определяющего программу, кнопка Изменить отображаться не будет.
- `NoRepair` — данный параметр `DWORD`-типа определяет, будет ли для соответствующей программы в списке апплета Установка и удаление программ отображаться кнопка Восстановить. Если значение данного параметра равно `1`, то для элемента списка, определяющего программу, кнопка Восстановить отображаться не будет.

ВНИМАНИЕ

Несмотря на то, что кнопка Удалить скрыта из диалога Установка и удаление программ, программу по-прежнему можно будет удалить с помощью команды rundll32.exe appwiz.cpl, WOW64Uninstall_RunDLL ,,,«название раздела программы в ветви реестра HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall».

Другие возможности

Здесь будут рассмотрены другие параметры операционной системы Windows XP, которые не используются для настройки конфигурации значков или окон Windows, но тем не менее могут быть вам интересны.

- `HKEY_CURRENT_USER\Control Panel\Desktop` — довольно интересной возможностью, которую позволяет выполнить операционная система Windows XP, является вывод сведений о версии Windows XP в правом нижнем углу экрана Рабочего стола. Для ее реализации достаточно присвоить `DWORD`-параметру `PaintDesktopVersion` значение, равное `1` (рис. 5.8).

  Другой возможностью, которую предоставляет пользователям операционная система Windows, является вывод обоев Windows, начиная с определенной точки экрана. Для этого используются два параметра строкового типа `WallpaperOriginX` и `WallpaperOriginY`. Первый из них определяет смещение левой стороны рисунка обоев от левой стороны экрана. Второй же параметр определяет смещение верхней стороны рисунка обоев от верхней стороны экрана.

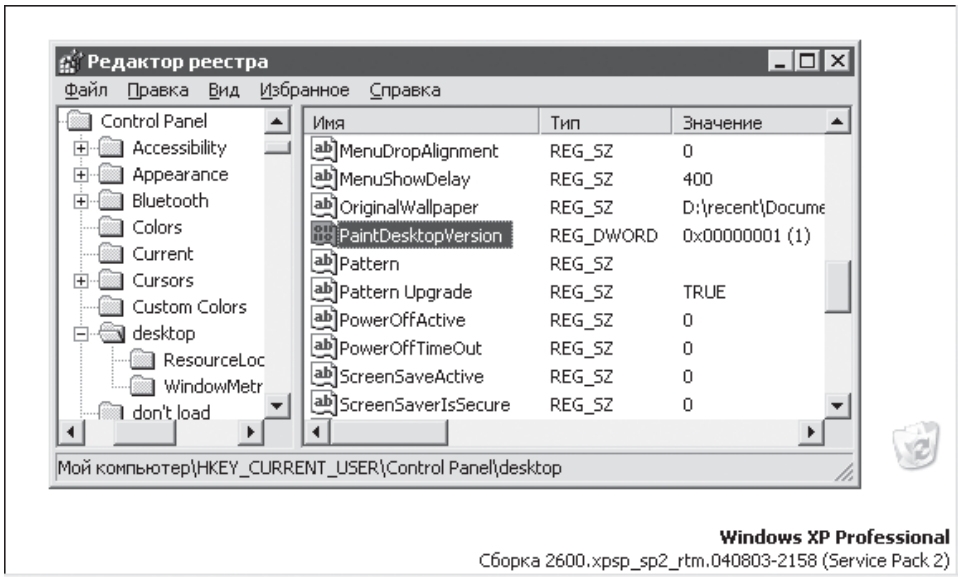


Рис. 5.8. Отображение версии Windows в правом нижнем углу экрана

С помощью данной ветви реестра можно проделать другой трюк — изменение количества строк и столбцов в выводимом после нажатия комбинации клавиш Alt+Tab диалоге. Для этого применяются два параметра строкового типа `CoolSwitchRows` и `CoolSwitchColumns`. Первый из них определяет количество строк в диалоге, а второй — количество столбцов. Например, на рис. 5.9 можно увидеть результат присвоения этим параметрам значений 3 и 3 (чтобы изменения вступили в силу, необходима перезагрузка компьютера или выход из системы — обычный перезапуск оболочки Windows не поможет).

Рис. 5.9. Изменение количества строк и столбцов диалога

- `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\AppKey` — содержит команды, которые будут выполняться при нажатии дополнительных клавиш клавиатуры, таких, например, как My Computer, Calculator, E-mail, Stop и т. д. Не секрет, что не всегда функциональная возможность данных клавиш может быть необходима, например, если у вас нет доступа к Интернету, зачем вам такие клавиши быстрого доступа, как Refresh, Forward,

Back, Home? Поэтому предлагаю переопределить команды, вызываемые соответствующими клавишами, чтобы они вызывали вашу любимую игру или программу.

Для этого необходимо определить формат содержимого приведенной выше ветви реестра — в ней хранятся разделы, названия которых определяют числовые идентификаторы, присвоенные клавишам быстрого доступа. А уже в этих разделах располагается параметр строкового типа `ShellExecute` — именно в нем находится команда, которая будет выполняться при нажатии соответствующей клавиши быстрого доступа.

Структура ветви реестра рассмотрена. Осталось только определить идентификаторы клавиш и названия самих клавиш, которые им соответствуют:

- `1` — Back (определяет кнопку Назад в окне браузера);
- `2` — Forward (указывает кнопку Вперед в окне браузера);
- `3` — Refresh (определяет кнопку Обновить в окне браузера);
- `4` — Stop (указывает кнопку Стоп в окне браузера);
- `5` — Search (аналогична комбинации клавиш Windows+F);
- `6` — Favorites (определяет кнопку Избранное в окне браузера);
- `7` — Home (указывает кнопку Домой в окне браузера);
- `8` — Mute (отключает звук в проигрывателе, назначение данной клавиши переопределить нельзя);
- `15` — E-mail (аналогична вызову программы Outlook Express);
- `16` — Media (аналогична вызову программы, ассоциированной с расширением CDA (по умолчанию Проигрыватель Windows Media));
- `17` — My Computer (аналогична вызову окна Мой компьютер);
- `18` — Calculator (аналогична вызову программы calc.exe).

Например, чтобы присвоить дополнительной клавише Back команду, открывающую вашу папку с играми, допустим D:\Games, нужно присвоить параметру `ShellExecute`, расположенному в ветви реестра `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\AppKey\1`, значение, равное `explorer /root, d:\games`.

ПРИМЕЧАНИЕ

После изменения команд некоторых функциональных клавиш у них появляется двойной набор команд. Чтобы показать это на примере, достаточно изменить команду клавиши, открывающей программу Outlook Express. Теперь если вы нажмете эту клавишу при установке фокуса на окне какой-либо папки, то по-прежнему откроется окно Outlook Express. Но если вы нажмете клавишу, когда фокус установлен на Рабочем столе или окне другой программы, то выполнится та команда, которую вы заложили в эту клавишу с помощью ветви реестра HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\AppKey.

- HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\App Paths — определяет все те команды, которые вы вводите в диалоге Запуск программы, чтобы открыть необходимую программу, расположенную в каталогах, отличных от %systemroot%\system32 и %systemroot%. Например, если у вас на компьютере установлен графический пакет Adobe Photoshop, то для его вызова из диалога Запуск программы вы используете команду Photoshop. И, несмотря на то, что, скорее всего, данный пакет установлен в одном из разделов ветви %programfiles%, команда Photoshop все равно открывает его окно.

  Такой феномен возможен лишь потому, что в приведенной выше ветви реестра существует раздел, имеющий название photoshop.exe (все разделы приведенной ветви реестра должны оканчиваться расширением EXE, иначе определяемые ими псевдонимы команд работать не будут). В этом разделе существует параметр (Ïî óìîë÷àíèþ), хранящий путь к программе, которая будет запускаться после ввода соответствующего ей псевдонима в диалоге Запуск программы.

  Что вам дает знание данной ветви реестра? Ну, если пофантазировать, то можно изменить название раздела photoshop.exe на, допустим, ph.exe. Теперь вы сможете вызвать программу Photoshop вводом всего двух символов.

  Можно создать свой собственный псевдоним для любимой программы. Для этого нужно просто создать новый раздел с любым названием (не забывайте, что раздел должен оканчиваться расширением EXE) в ветви системного реестра HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\App Paths. После этого в параметре (Ïî óìîë÷àíèþ) созданного раздела нужно определить путь к своей любимой программе.

ПРИМЕЧАНИЕ

Некоторые программы, кроме создания псевдонима, могут содержать в приведенной выше ветви реестра и другие параметры. Например, в ветви системного реестра HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\App Paths\setup.exe (принадлежит программе установки других приложений) может находиться DWORD-параметр RunAsOnNonAdminInstall. Если его значение равно 1, то при каждом запуске программы Setup.exe для установки какой-нибудь программы пользователем, не имеющим на это привилегий, будет выводиться диалог Запуск от имени, в котором можно зарегистрироваться под учетной записью, имеющей права на установку программ в системе.

Точно такой же параметр может храниться в ветви реестра HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\App Paths\winnt32.exe, а также в ветви HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\App Paths\install.exe.

Есть и еще один общий для всех разделов параметр строкового типа — RunAsCommand. Если он присутствует в разделе и его значение равно, допустим, 0, то ни один пользователь не сможет запустить программу, определяемую данным псевдонимом. Имеется в виду, что не сможет запустить не только из диалога Запуск программы, но и с помощью ярлыков или самого файла программы (можно только с помощью консоли cmd.exe).

- `HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\Advanced` — может содержать один интересный параметр `DWORD`-типа, который называется `NewDragImages`. Он может принимать значения `0` или `1`. В зависимости от используемого значения вы сможете увидеть одно из изображений, приведенных на рис. 5.10 (рисунок слева создан при копировании пункта меню Пуск на Рабочий стол с использованием значения параметра, равного `1`, а рисунок справа создан при использовании значения, равного `0`).

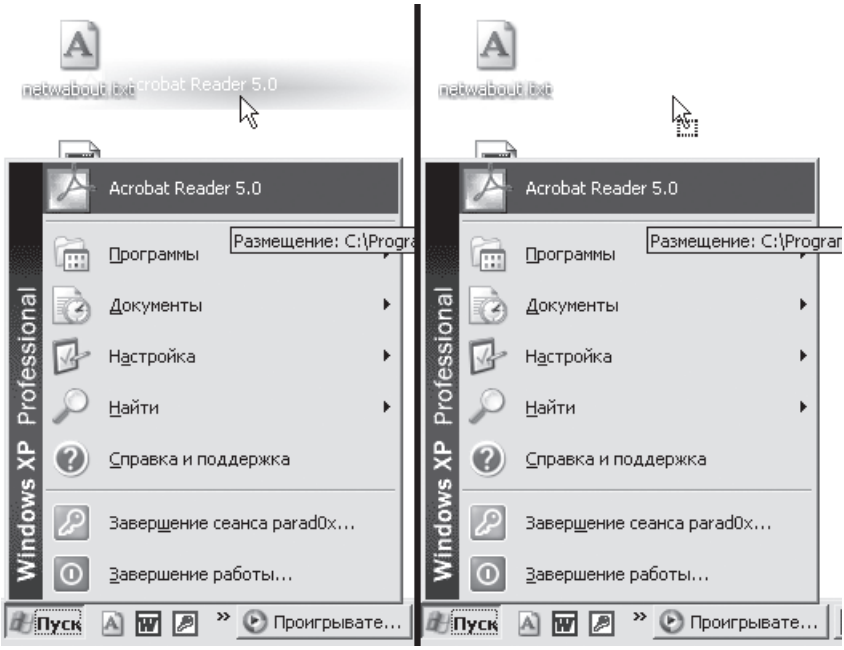


Рис. 5.10. Изменение способа отображения копируемого файла

- `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\MyComputer` — мы с вами уже рассматривали содержимое раздела `NameSpace` этой ветви реестра, но, кроме него, ветвь корневого раздела `HKEY_LOCAL_MACHINE` имеет еще три раздела: `BackupPath`, `cleanuppath` и `DefragPath`. Все эти ветви включают в себя параметр `(Ïî óìîë÷àíèþ)`, значение которого определяет команду, выполняющуюся при нажатии той или иной кнопки на вкладке Сервис диалога Свойства: Диск (параметр первого раздела определяет команду кнопки Выполнить архивацию, второго — Выполнить проверку, а третьего — Выполнить очистку).

  Например, если присвоить параметру `(Ïî óìîë÷àíèþ)` раздела `BackupPath` значение, допустим, `cmd.exe`, то после нажатия кнопки Выполнить архивацию перед вами отобразится командный процессор `cmd.exe`. Конечно, пример с программой `cmd.exe` не совсем удачен, но если вы используете для перечисленных выше действий программы сторонних производителей, то именно их вы-

зов можно присвоить параметрам (Ïî óìîë÷àíèþ) разделов BackupPath, cleanuppath и DefragPath.

- HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\FindExtensions\Static — определяет содержимое пункта Найти меню Пуск. Оно строится в следующем формате: сначала идет набор разделов, каждый из которых определяет одну команду меню Пуск. Например, в приведенной ветви реестра могут находиться следующие разделы:
  - ShellSearch — определяет команду Файлы и папки пункта Найти меню Пуск;
  - WabFind — задает команду Людей пункта Найти меню Пуск;
  - WebSearch — определяет команду В Интернете пункта Найти меню Пуск.

  Параметр (Ïî óìîë÷àíèþ) каждого из этих разделов определяет ActiveX-объект, который обрабатывает выбор соответствующей команды из пункта Найти меню Пуск. Кроме этого параметра, разделы могут содержать другие разделы, названия которых определяются в формате 0, 1, 2, 3. Параметр (Ïî óìîë÷àíèþ) каждого из этих разделов определяет соответствующую строку названия (если в разделе не существует параметра строкового типа LocalizedString, значение которого, как известно, всегда переопределяет параметр (Ïî óìîë÷àíèþ)). При этом только значение параметра (Ïî óìîë÷àíèþ) раздела 0 может отображаться в качестве названия команды пункта Найти меню Пуск. Каждый из разделов 0, 1, 2..., в свою очередь, должен содержать подраздел, который называется DefaultIcon. Параметр (Ïî óìîë÷àíèþ) этого подраздела определяет значок, отображаемый напротив названия команды пункта Найти меню Пуск.

  Зачем же все это нужно? Во-первых, для того, чтобы удалить ненужные команды пункта Найти меню Пуск. Во-вторых, что уже интересней, чтобы добавить свои команды к данному меню. Например, попробуйте добавить к данному меню команду вызова апплета Установка и удаление программ. Для этого нужно создать раздел с любым именем в ветви реестра HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\FindExtensions\Static и его параметру (Ïî óìîë÷àíèþ) присвоить значение {2559A1F7-21D7-11D4-BDAF-00C04F60B9F0}. После этого нужно создать дочерний раздел 0 и его параметру (Ïî óìîë÷àíèþ) присвоить строку названия, например незамысловатое Установка и удаление программ. И наконец, в параметре (Ïî óìîë÷àíèþ) подраздела DefaultIcon, который должен быть создан в дочернем разделе, нужно определить путь к файлу рисунка, используемого в качестве значка (рис. 5.11).

- HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\FileAssociation — раз уж чуть выше речь зашла о меню Пуск, то поговорим еще о двух параметрах строкового типа, предназначенных для выполнения конфигурации нового меню Пуск, — AddRemoveApps и AddRemoveNames. Первый из них определяет имена программ (пишутся через точку с запятой),

которые не будут помещаться в новое меню Пуск, сколько бы их ни вызывали. Второй же содержит части названия имен файлов, ссылки на которые не будут помещаться в новое меню Пуск, сколько бы их ни вызывали.

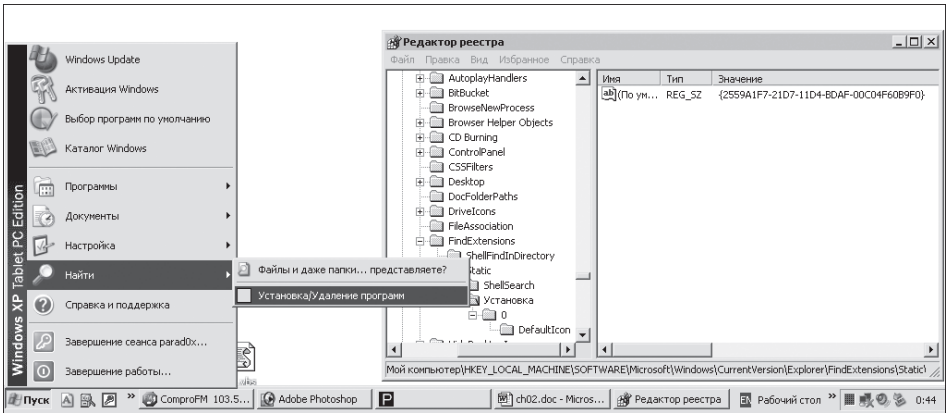


Рис. 5.11. Создание своей команды пункта Найти меню Пуск

- `HKEY_CURRENT_USER\Software\Microsoft\Java VM` — еще одна интересная ветвь реестра, параметры которой определяют настройки консоли Java. К контексту данной главы эти параметры не очень подходят. Все, кроме одного, имеющего название `EnableJavaConsole`. Параметр имеет тип `REG_BINARY`, и если его значение будет равно `1`, то в своем меню Вид Проводника и браузера Internet Explorer вы сможете встретить команду Окно языка Java, после нажатия которой перед вами предстанет окно, подобное изображенному на рис. 5.12. Автор не очень разбирается в языке Java, поэтому судить о важности данной команды предоставляет читателям.
- `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon` — используется для хранения установок сервиса `WINLOGON` — именно ему мы доверяем свой логин и пароль при входе в систему. Конечно, установки `WINLOGON` малопригодны для настройки оболочки, но все-таки существует один параметр, который подходит к описываемой теме. Этот параметр имеет строковый тип и называется `Background`. Его значение определяет цвет фона, который будет использоваться для отображения Рабочего стола при выводе диалога регистрации в системе (только если используется классический вход в систему), и имеет уже знакомый вам RGB-формат. Например, если присвоить этому параметру значение `000`, то вместо стандартного голубоватого фона вы получите строгий черный.
- `HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer` — содержит очень интересный параметр, имеющий `DWORD`-тип, — `MaximizeApps`. Если его значение будет равно `1`, то все запускаемые вами программы будут открываться на весь экран. Предположительно этот параметр применяется для отображения на весть экран командной строки `cmd.exe` при

использовании альтернативного входа в систему Безопасный режим с использованием командной строки.

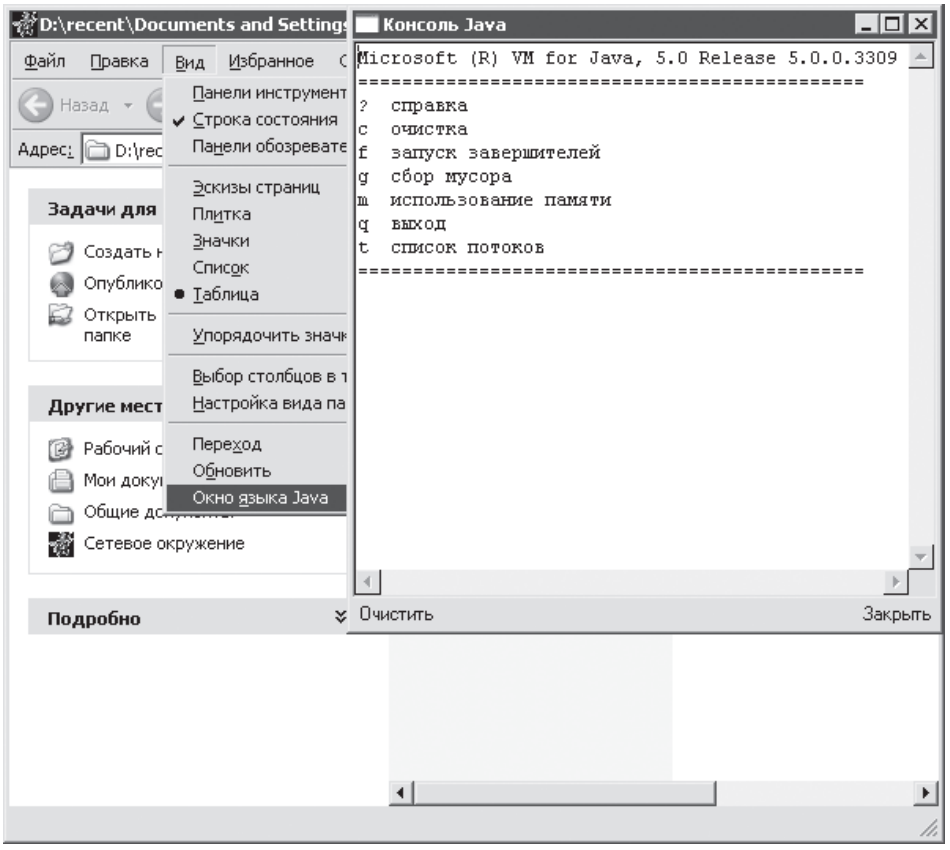


Рис. 5.12. Добавление к меню Вид вызова консоли языка Java

- `HKEY_CURRENT_USER\Software\Microsoft\Windows NT\CurrentVersion\Network\Persistent Connections` — содержит параметр строкового типа `SaveConnections`. Его значение определяет, будет ли операционная система по умолчанию создавать перманентные подключения к сетевым папкам и принтерам (перманентные подключения — это подключения, которые будут восстанавливаться после следующего входа пользователя в систему). Если значение равно `no`, то перманентные подключения создаваться не будут. По умолчанию значение равно `yes`, хотя иногда это может нарушить систему безопасности (например, когда в системе используются скрытые сетевые ресурсы, заканчивающиеся на знак `$` и при этом администратор не хотел бы, чтобы пользователи знали о таких сетевых ресурсах).
- `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\NetworkProvider` — может содержать `DWORD`-параметр `RestoreConnection`.

По умолчанию его значение равно 0, что говорит о том, что Windows разрешено использовать фантомные подключения. Если значение равно 1, то это будет запрещено.

ПРИМЕЧАНИЕ

Фантомные подключения — это перманентные подключения, которые при входе пользователя в систему считаются восстановленными, хотя фактически система не выполняла их восстановление. Плюсом фантомных подключений является то, что такие подключения при входе пользователя физически не восстанавливаются, поэтому вход будет быстрее. К тому же если удаленный компьютер, доступ к которому система должна восстановить, в данный момент выключен, то система при входе пользователя не будет выводить диалог о недоступности компьютера (при отключении фантомных подключений операционная система будет выдавать сообщение о том, что она не смогла восстановить подключение, если необходимый компьютер недоступен). Минусом же такого подключения является небольшая задержка перед его использованием в первый раз, необходимая чтобы система смогла физически установить подключение удаленного компьютера.

- `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\lanmanserver\Shares` — раз уж зашла речь о сети, то немного продолжим эту тему. В предыдущей части книги упоминалась команда (даже несколько команд), с помощью которой можно было создать общедоступную папку. Но это не единственная возможность операционной системы, предназначенная для работы с общедоступными папками, ведь вы уже знаете, что большую часть всей своей конфигурационной информации Windows хранит в реестре. Информация о общедоступных папках не является исключением — именно для ее хранения и применяется описываемая ветвь реестра.

  Ветвь включает в себя список параметров `REG_MULTI_SZ`-типа, каждый из которых описывает один общедоступный ресурс (имя параметра равно названию общедоступной папки) (рис. 5.13). Описание ресурса состоит из следующих строк (находятся в значении параметра):

  - `CSCFlags` — определяет флаги кэширования папки для работы в автономном режиме (например, если данный параметр равен `48`, то кэширование выполняться не будет, если равен `16`, то будет выполняться кэширование документов в автоматическом режиме, если равен `0`, то кэширование будет выполняться в ручном режиме);
  - `MaxUses` — определяет количество пользователей, которые могут одновременно получить доступ к ресурсу (если определено неограниченное количество пользователей, то эта строка примет вид `MaxUses=4294967295`);
  - `Permissions` — назначение неизвестно (всегда равен `0`);
  - `Remark` — определяет примечание к общедоступному ресурсу (оно создается с помощью поля Примечание на вкладке Доступ);
  - `Type` — назначение неизвестно (для папок всегда равен `0`, а для принтеров равен `1`).

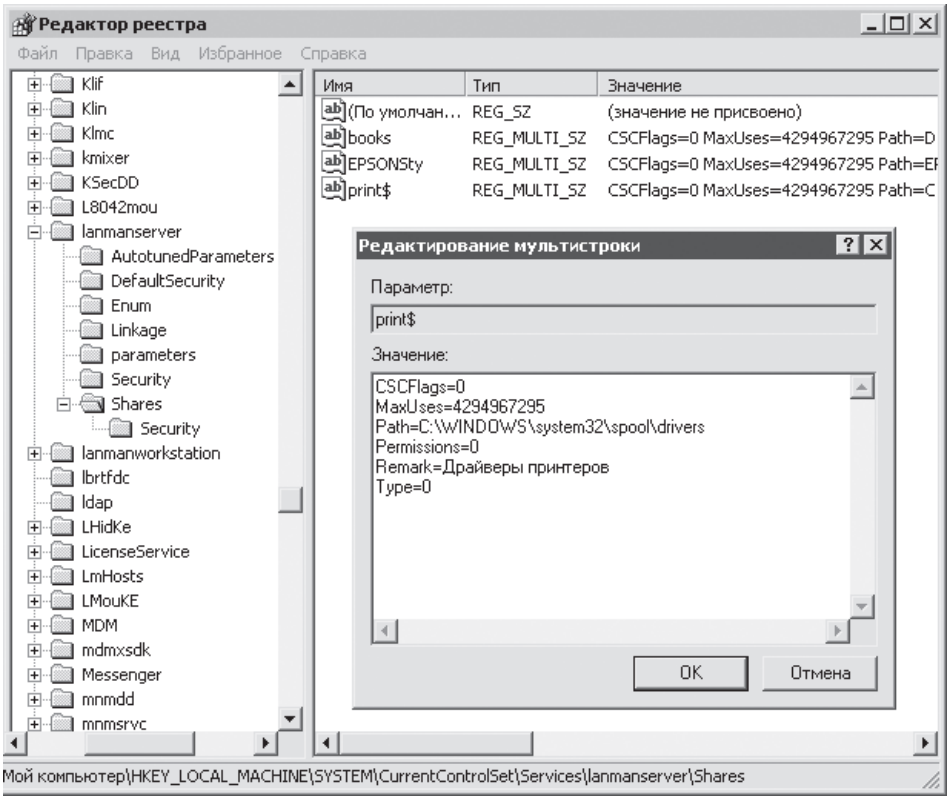


Рис. 5.13. Хранение сведений о папках, к которым открыт общий доступ

Имейте в виду, что файлы и папки, находящиеся в общем доступе, менее защищены, чем при отсутствии общего доступа к ним. По этой причине рекомендуется периодически проверять содержимое таких папок.

Глава 6

Internet Explorer и Outlook Express

- **Internet Explorer**
- **Outlook Express**
- **Скрытие возможности работы с Windows Messenger**

Уже были рассмотрены настройки оболочки Windows, доступ к которым нельзя получить с помощью диалоговых окон операционной системы Windows. Был также рассмотрен формат хранения в реестре информации о расширениях файлов и ActiveX-объектов. Сейчас же опишем подобные настройки Internet Explorer и Outlook Express. Иначе говоря, те настройки, доступ к которым нельзя получить с помощью диалоговых окон данного браузера или почтового клиента, но можно — с помощью реестра Windows XP.

Internet Explorer

Первой программой, настройки которой будут рассмотрены, станет стандартный браузер для просмотра страниц Интернета, входящий в поставку операционной системы Windows XP, — Internet Explorer. Как говорилось раньше, в поставку Windows XP входит браузер Internet Explorer версии 6.0 (версию браузера можно посмотреть в параметре строкового типа `Version` из ветви системного реестра `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Internet Explorer`).

Параметры браузера хранятся в нескольких ветвях реестра. Основные пользовательские параметры оболочки браузера Internet Explorer расположены в ветви реестра `HKEY_CURRENT_USER\Software\Microsoft\Internet Explorer`. Настройки самого браузера, не зависящие от конкретного пользователя, который находится в данный момент в системе, хранятся в ветви системного реестра `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Internet Explorer`. Параметры конфигурации браузера Internet Explorer для текущего пользователя расположены в ветви реестра `HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Internet Settings`. Параметры конфигурации браузера для всех пользователей — в ветви реестра `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings`.

Оболочка

На самом деле настроек браузера, доступ к которым нельзя получить с помощью диалоговых окон, не очень много, поэтому глава будет небольшой. Но она будет, и начнется, как и все другие главы книги, с рассмотрения настроек оболочки.

В рассказе о настройках оболочки Windows упоминалось о способе, используемом для изменения фона панели инструментов Проводника. То же самое можно сделать и для панели инструментов браузера Internet Explorer и почтового клиента Outlook Express. Для этого также применяется параметр строкового типа из ветви реестра `HKEY_CURRENT_USER\Software\Microsoft\Internet Explorer\Toolbar`. Эта ветвь содержит два параметра строкового типа, определяющих фон панели инструментов браузера Internet Explorer, — `BackBitmap` и `BackBitmapIE5`. Если вы будете использовать первый из этих параметров для указания пути к рисунку фона панели инструментов, то фон изменится не только в браузере, но и в почтовом клиенте Outlook Express и Проводнике Windows (если фон для Проводника не был

переопределен параметром `BackBitmapShell`). Если же вы будете использовать второй параметр, то указанный рисунок фона будет применяться только для отображения на панели инструментов браузера Internet Explorer.

В разделе о параметрах оболочки Windows рассказывалось, как изменить логотип, используемый в Проводнике и отображаемый в верхнем правом углу окна. То же самое можно сделать и для браузера Internet Explorer (а также для почтового клиента Outlook Express) — для этого применяется ветвь реестра `HKEY_CURRENT_USER\Software\Microsoft\Internet Explorer\Toolbar`. Если для настройки логотипов Проводника использовались параметры `SHSmallBitMap` и `SHBigBitMap`, то для настройки логотипов браузера Internet Explorer применяются параметры строкового типа `BrandBitMap` и `SmBrandBitMap`. Параметр `BrandBitMap` определяет путь к файлу изображения (BMP), который будет использоваться при нормальном отображении окна браузера (не в полноэкранном режиме). Параметр `SmBrandBitMap` определяет изображение, которое применяется в полноэкранном отображении браузера (его можно вызвать нажатием клавиши F11).

Изменить логотип можно и с помощью параметров строкового типа `BigBitmap` и `SmallBitmap`, расположенных в этой же ветви реестра, — они переопределяют собой параметры `BrandBitMap` и `SmBrandBitMap`. Пример изменения параметров логотипа приведен на рис. 6.1.

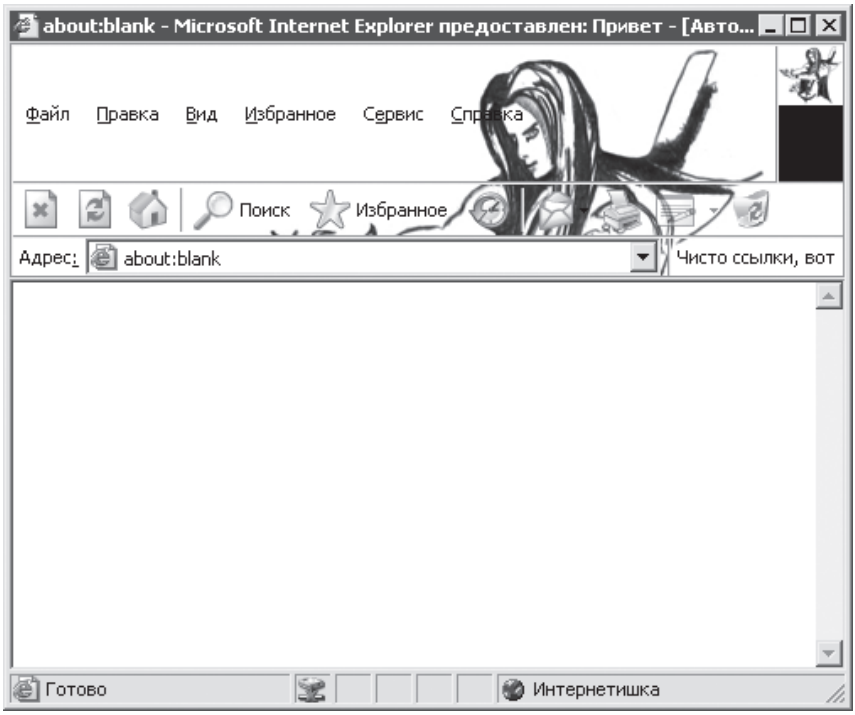


Рис. 6.1. Изменение интерфейса браузера

ПРИМЕЧАНИЕ

Если при изменении логотипа в Проводнике ничего не говорилось о размерах, так как Проводник автоматически изменял размеры файла рисунка (если его размер был в пределах 50 пикселов), когда он был слишком большим, то при изменении логотипа браузера Internet Explorer — стоит сказать. Параметр BrandBitMap должен указывать на изображение размером 38 × 38 пикселов, а параметр SmBrandBitMap — на изображение размером 22 × 22 пиксела. Если изображение будет больше, то браузер его обрежет.

Еще одним элементом, который можно настроить в браузере Internet Explorer, является название ссылок, отображаемое напротив адресной строки панели инструментов (на рис. 6.1 показан пример, где соответствующий строковый параметр имеет значение, равное `×èñòî ññûëêè, âîò`). Для этого применяется параметр строкового типа `LinksFolderName`, который содержит имя списка для ссылок и находится в ветви реестра `HKEY_CURRENT_USER\Software\Microsoft\Internet Explorer\Toolbar`.

ПРИМЕЧАНИЕ

При изменении значения параметра LinksFolderName в меню Избранное автоматически создается папка с указанным в параметре именем — ссылки именно из этой папки будут содержаться в списке напротив адресной строки.

Можно также добавить к заголовку браузера произвольный текст. Для этого предназначен параметр строкового типа Window Title, расположенный в ветви реестра HKEY_CURRENT_USER\Software\Microsoft\Internet Explorer\Main.

Вы хотите изменить имя зоны, отображаемое в правом нижнем углу окна браузера? Это тоже можно сделать. Например, можно изменить имя зоны Интернет на имя Интернетишка (см. рис. 6.1). Для реализации этого трюка понадобится всего одна ветвь реестра — `HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Internet Settings\Zones\3`. Она хранит настройки уровня безопасности зоны Интернета, среди которых находится параметр строкового типа `DisplayName`. По умолчанию его значение равно `Èíòåðíåò`, но вы можете изменить его на то, которое захотите видеть в браузере. Можно также изменить и названия других зон — все они содержатся в параметрах строкового типа `DisplayName` различных разделов, дочерних по отношению к разделу `HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Internet Settings\Zones`. Например, к таким разделам относятся следующие: `0` — зона Мой компьютер; `1` — Местная интрасеть; `2` — Надежные узлы; `3` — Интернет; `4` — Ограниченные узлы.

Кроме названий зон Интернета и папок ссылок, реестр Windows позволяет изменить названия программ в полях на вкладке Программы диалога Свойства обозревателя (рис. 6.2). Все эти названия находятся в параметрах `(Ïî óìîë÷àíèþ)`

разделов ветви реестра HKEY_LOCAL_MACHINE\SOFTWARE\Clients. Эта ветвь включает в себя разделы, каждый из которых идентифицирует программы одного типа. Например, раздел Mail идентифицирует программы, которые можно выбрать в поле Электронная почта вкладки Программы. В этих разделах содержатся вложенные подразделы, каждый из которых определяет одну программу, которую можно выбрать в соответствующем поле на вкладке Программы. Параметр (Ïî óìîë÷àíèþ) этих подразделов как раз и определяет название программы, которое будет отображаться в списке соответствующего поля. Например, чтобы изменить название программы Outlook Express поля Электронная почта, нужно воспользоваться параметром (Ïî óìîë÷àíèþ) из ветви реестра HKEY_LOCAL_MACHINE\SOFTWARE\Clients\Mail\Microsoft Outlook.

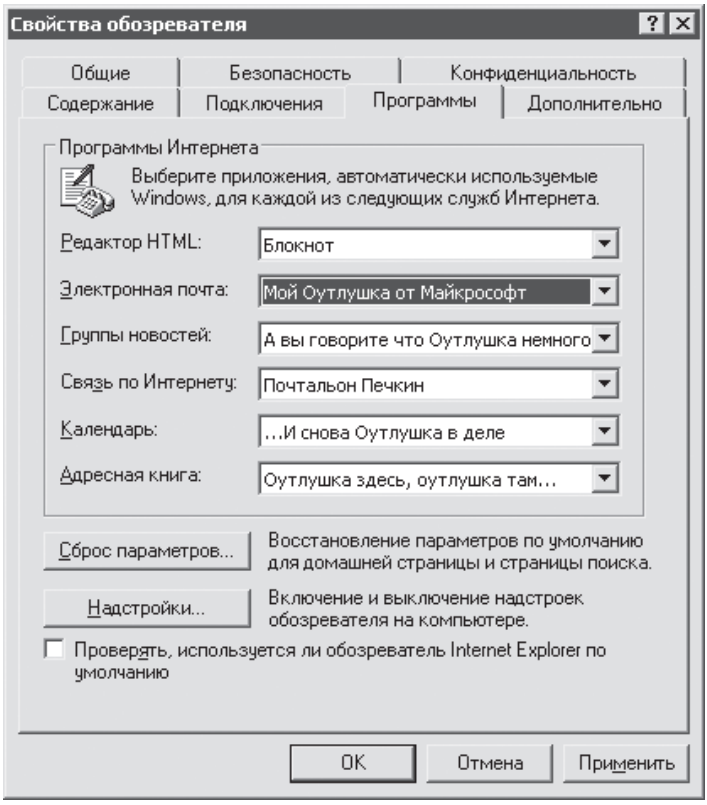


Рис. 6.2. Изменение названий программ

ПРИМЕЧАНИЕ

Если вы измените название программы Outlook Express в поле Электронная почта, то оно будет использоваться и в новом меню Пуск для идентификации ссылки на почтовый клиент Outlook Express.

А как вам возможность добавления к панели инструментов или меню Сервис своей команды? Это также можно сделать с помощью реестра — достаточно воспользоваться ветвью реестра HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Internet Explorer\Extensions. Она хранит разделы, названия которых являются GUID-номерами. Каждый из них определяет одну дополнительную клавишу или команду меню Сервис и может содержать следующие параметры строкового типа.

- ButtonText — определяет строку подсказки, которая будет отображаться при наведении и удержании указателя мыши над созданной кнопкой.
- Clsid — его значение должно быть равно {1FBA04EE-3024-11D2-8F1F-0000F87ABD16}, в противном случае кнопка создана не будет.
- Default Visible — определяет, будет ли отображаться на панели инструментов создаваемая вами кнопка (если значение равно YES, то кнопка будет отображаться, если же значение равно NO, то не будет).
- Exec — указывает команду, которая будет выполняться при нажатии создаваемой кнопки (при этом следует учитывать, что в разделе может присутствовать либо параметр Script, либо параметр Exec).
- HotIcon — определяет значок, на который будет изменяться стандартный значок вашей кнопки при наведении на нее указателя мыши (значок для кнопки должен находиться в библиотеке).
- Icon — указывает путь к значку, который будет использоваться для отображения создаваемой вами кнопки (значок для кнопки должен находиться в библиотеке).
- MenuStatusBar — определяет текст, отображаемый в строке статуса при выборе созданной вами команды из меню Сервис.
- MenuText — указывает название создаваемой вами команды в меню Сервис.
- Script — определяет путь к сценарию (HTML-файлу), который будет выполняться при нажатии вашей кнопки или выборе из меню Сервис ее аналога (при этом следует учитывать, что в разделе может присутствовать либо параметр Script, либо параметр Exec).

Вот и все параметры, которые могут присутствовать в разделе, формирующем кнопку на панели инструментов или команду в меню Сервис. Теперь для примера попробуем создать свою кнопку. Результат ее создания можно увидеть на рис. 6.3, а листинг REG-файла, описывающего созданные параметры, приведен ниже.

ПРИМЕЧАНИЕ

Кнопка не всегда сразу отображается на панели инструментов после того, как вы ее создали. Иногда необходимо добавить ее на панель инструментов или включить с помощью диалога, вызываемого командой Управление надстройками меню Сервис (была создана надстройка для браузера Internet Explorer).

6.1.

```
Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Internet Explorer\Extensions\{27A88317-
08F0-4068-A8B3-7FAB3255C4BC}]
"clsid"="{1FBA04EE-3024-11D2-8F1F-0000F87ABD16}"
"Default Visible"="yes"
"ButtonText"="
"Icon"="shell32.dll,32"
"HotIcon"="shell32.dll,33"
"MenuText"="
"MenuStatusBar"="                e:\\music"
"Exec"="e:\\music"
```

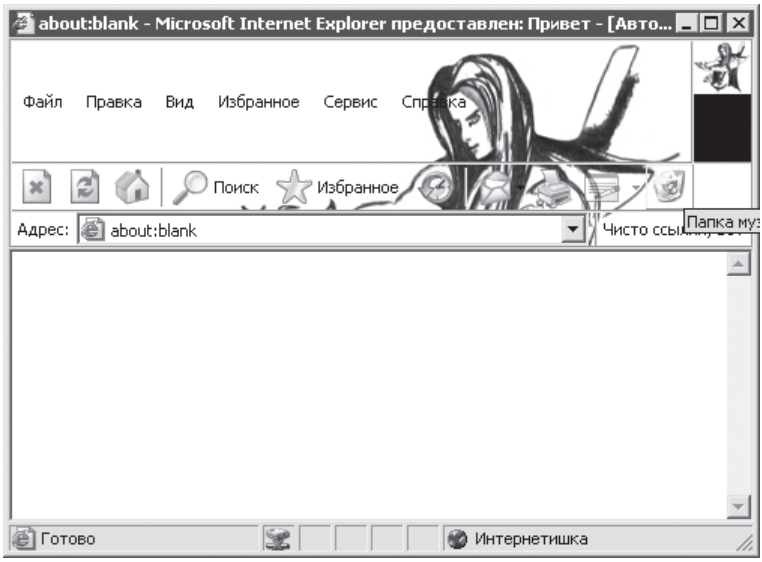


Рис. 6.3. Создание кнопки на панели инструментов

Конфигурация

Теперь несколько слов будет сказано о параметрах браузера Internet Explorer, которые изменяют настройки конфигурации браузера или способ его подключения к Интернету. Например, к ним можно отнести параметр строкового типа `Download Directory`. Он находится в ветви реестра `HKEY_CURRENT_USER\Software\Microsoft\Internet Explorer` и определяет путь к папке, в которую по умолчанию будут закачиваться файлы из Интернета. По умолчанию значение этого параметра определяет путь к папке Рабочего стола пользователя, что в некоторых случаях может быть неудобно.

Фиксация расположения окна браузера

Другой возможностью, которую предоставляет реестр Windows, является возможность запрета сохранения настроек высоты и ширины окна браузера, а также его расположения на экране. Самым простым способом, с помощью которого это можно сделать, является ограничение доступа к ветви реестра `HKEY_CURRENT_USER\Software\Microsoft\Internet Explorer\Main`. Но данный способ имеет один большой недостаток (конечно, в некоторых случаях это достоинство) — будет также запрещено изменять многие настройки браузера Internet Explorer, доступ к которым можно получить с помощью вкладки Дополнительно диалога Свойства обозревателя. Если вам не хочется запрещать доступ к этим настройкам, то можно воспользоваться более экстравагантным способом. Для его реализации понадобится DWORD-параметр `Window_Min_Height` из ветви реестра `HKEY_CURRENT_USER\Software\Microsoft\Internet Explorer\Main`. Достаточно присвоить этому параметру значение `0õ00111111`. После этого любые настройки расположения или размеров окна браузера, произведенные в последующие сеансы работы с Интернетом, не будут сохраняться в системе (исключением из правила является возможность установки полноэкранного режима (клавиша F11) — если при закрытии окна браузера использовался полноэкранный режим, то он будет использоваться и при следующем запуске браузера).

Программа просмотра HTML-кода сайта

С помощью реестра можно определить программу, в которой будет открываться HTML-код страницы после выбора из меню Вид команды Просмотр HTML-кода. Для этого достаточно параметру `(Ïî óìîë÷àíèþ)`, расположенному в ветви реестра `HKEY_CURRENT_USER\Software\Microsoft\Internet Explorer\Default HTML Editor\shell\edit\command`, присвоить значение содержащее путь к программе, которая будет использоваться для просмотра HTML-кода страницы.

Префикс по умолчанию

Можно также определить префикс, который по умолчанию будет подставляться к адресу в адресной строке, если он там явно не указан. По умолчанию используется префикс http://, который говорит браузеру о том, что он должен отослать запрос службе WWW. Но если вы чаще работаете с другими службами, например FTP, то можно указать другой префикс по умолчанию. Информация о возможных префиксах расположена в ветви реестра `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\URL`. Ветвь содержит два раздела — `DefaultPrefix` и `Prefixes`. Первый определяет префикс по умолчанию, который хранится в его параметре `(Ïî óìîë÷àíèþ)`. Например, по умолчанию в параметре хранится значение `http://`, но можно заменить это значение, допустим, значением `ftp://` — теперь, если пользователь не укажет префикс, будет автоматически подставляться префикс ftp://. Раздел `Prefixes`, в свою очередь, определяет соответствие префиксов различным типам адресов. Он хранит строковые параметры, имена которых определяют службу, а значения параметров — используемый ею префикс.

Создание команд для запроса к поисковым системам

И еще несколько слов об адресной строке. Вы часто пользуетесь такими поисковыми системами, как Rambler, Google или Yandex? И при этом вы сначала заходите на стартовую страницу поисковой системы, а потом уже указываете запрос для поиска? Тогда могу предложить вам более быстрый поиск — поиск непосредственно из адресной строки Internet Explorer. Другими словами, вы вводите запрос в адресной строке браузера, а браузер отправляет его поисковой системе и возвращает вам страницу результатов. Для реализации этого трюка понадобится ветвь реестра `HKEY_CURRENT_USER\Software\Microsoft\Internet Explorer\SearchUrl` — нужно создать в ней свой раздел, название которого вы и будете вводить в адресной строке перед запросом. Содержимое созданного раздела будет определять адрес поисковой системы, которой вы будете передавать свой запрос. Этот адрес нужно указать в параметре (`Ïî óìîë÷àíèþ`) созданного раздела. Например, можно указать адреса на следующие поисковые системы:

- Rambler — параметр (`Ïî óìîë÷àíèþ`) должен иметь следующее значение: `http://search.rambler.ru/srch?words=%s`;
- Yandex — `http://www.yandex.ru/yandsearch?text=%s`;
- Google — `http://www.google.ru/search?hl=ru&lr=lang_ru&q=%s`;
- база знаний Microsoft — такое значение: `http://support.microsoft.com/default.aspx?scid=kb;en-us;%s`.

Вот и все. Например, если вы присвоите параметру (`Ïî óìîë÷àíèþ`) ветви реестра `HKEY_CURRENT_USER\Software\Microsoft\Internet Explorer\SearchUrl\r` значение `http://search.rambler.ru/srch?words=%s`, то для поиска в поисковой системе Rambler страниц, содержащих слова Привет и Пока, нужно будет ввести следующий запрос в адресной строке браузера — `r Ïðèâåò Ïîêà`. После этого поисковая система вернет вам список сайтов, содержащих введенный вами запрос.

ПРИМЕЧАНИЕ

Раз уж была затронута тема адресной строки, то определим, как формируется список уже просмотренных вами адресов сайтов Интернета в этой самой адресной строке (если данная возможность не отключена). Для этого опять-таки применяется содержимое реестра — список просмотренных вами адресов Интернета (подобные списки еще называют списками MRU) находится в ветви системного реестра HKEY_CURRENT_USER\Software\Microsoft\Internet Explorer\TypedURLs. Ветвь содержит строковые параметры, каждый из которых имеет название, создаваемое в следующем формате: url1, url2, url3 и т. д. Значения этих параметров как раз и хранят вводимые вами адреса URL. Вы можете самостоятельно ввести необходимые адреса Интернета, которые используете наиболее часто, а потом отредактировать права на данную ветвь реестра, оставив для себя только права на чтение содержимого ветви, чтобы случайно не заменить необходимые вам адреса URL адресной строки.

Создание синонимов к адресам адресной строки

Другой возможностью, которую предоставляет пользователям браузер Internet Explorer, является возможность создания синонимов к адресам Интернета. Например, гораздо легче ввести в адресной строке что-то вроде `about:vasia`, чем `http://www.vasia_super_cite/index.htm?passw=sss&login=ddd`. Да, вы уже догадались, что та же строка `about:blank`, используемая для отображения пустой страницы, является синонимом. При этом та команда, в которую транслируется строка `about:blank` при поиске сайта, записана в ветви системного реестра `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Internet Explorer\AboutURLs`. Ветвь содержит строковые параметры, имена которых являются строками, записываемыми после `about:` для идентификации синонима. Значения же параметров являются истинными адресами, в которые будут транслироваться синонимы. Например, значение параметра, идентифицирующего такую строку `about:blank`, равно `res://mshtml.dll/blank.htm` (рис. 6.4). Вы и сами можете добавлять в раздел `AboutURLs` свои параметры.

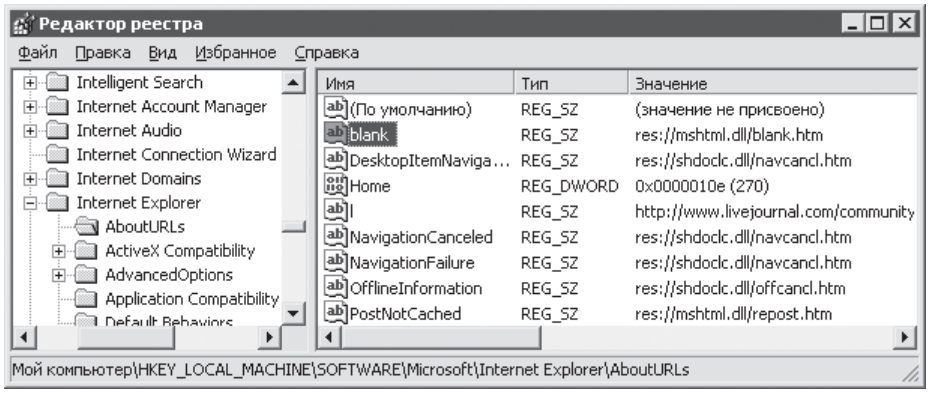


Рис. 6.4. Хранение синонимов адресов Интернета

Если же синонимы создавать не хочется, но у вас имеется один особо любимый сайт, то адрес к нему можно зафиксировать за кнопкой С исходной на вкладке Общие диалога Свойства обозревателя. После нажатия этой кнопки в качестве стартовой страницы устанавливается адрес того сайта, который за ней зафиксирован. По умолчанию адресом, который зафиксирован за кнопкой С исходной, является адрес сайта Microsoft (http://www.microsoft.com/isapi/redir.dll?prd=ie&pver=6&ar=msnhome), но если вы никогда не пользуетесь этим сайтом, то можете смело изменить значение параметра строкового типа `Default_Page_URL` из ветви системного реестра `HKEY_CURRENT_USER\Software\Microsoft\Internet Explorer\Main`.

Определения максимального количества выводимых браузером символов

Существует еще одна интересная возможность — указание запрета вывода строк текста больше определенного количества символов. Другими словами, можно указать

количество символов в строке текста, после которого остальные символы строки будут обрезаться. Для этого применяется DWORD-параметр `MaxRenderLine` из ветви реестра `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Internet Explorer\ Main`, который и определяет возможное количество символов. Например, на рис. 6.5 приведены три окна браузера Internet Explorer — в левом окне отображается ситуация, когда значение параметра `MaxRenderLine` равно `0`, в среднем окне — когда значение равно `2`, а в правом окне значение равно `400`.

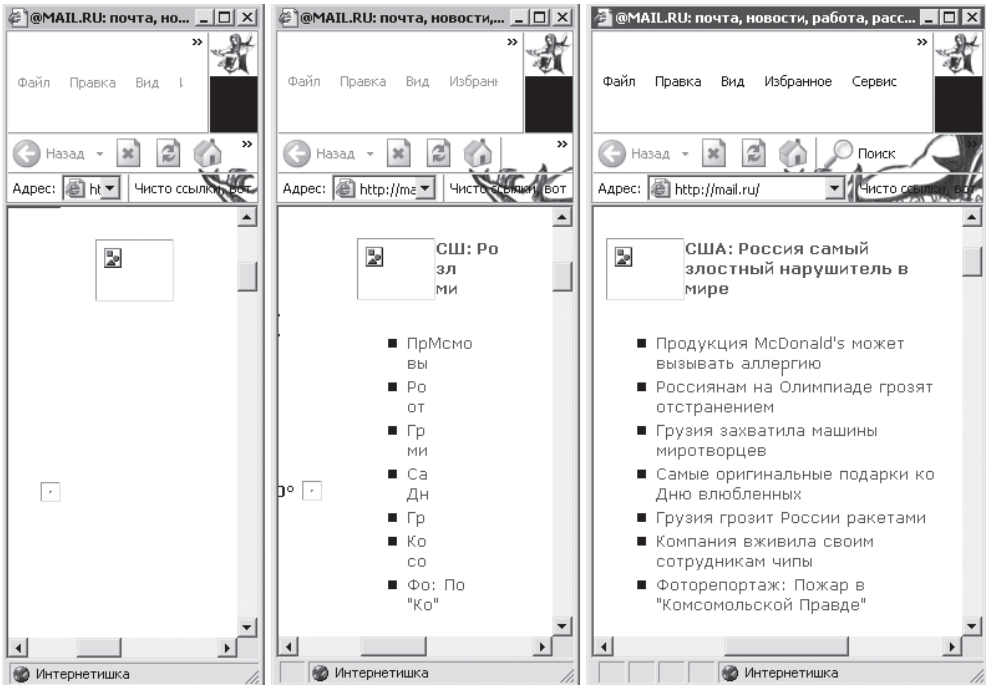


Рис. 6.5. Изменение максимальной длины строки текста

Разрешение выполнения ActiveX-объектов на локальном компьютере

Еще одной интересной возможностью, которую можно отредактировать с помощью реестра, является слежение системы за ActiveX-объектами при работе с HTML-страницами на локальном компьютере. Если вы часто просматриваете HTML-страницы, находящиеся на жестком диске вашего компьютера, то должны были заметить, что по умолчанию система запрещает выполнение ActiveX-объектов при работе с HTML-страницей, сохраненной на жестком диске, что иногда может мешать работе HTML-страницы. Если вы полностью доверяете создателям страницы, то можно отключить данный запрет системы. Для этого используется ветвь реестра `HKEY_CURRENT_USER\SOFTWARE\Microsoft\Internet Explorer\ Main\FeatureControl\FEATURE_LOCALMACHINE_LOCKDOWN`. Чтобы система не запрещала ActiveX-объекты, достаточно в этой ветви создать DWORD-параметр `iexplore` и присвоить ему значение, равное `0`. В этой ветви можно также создать

DWORD-параметр * — он будет определять состояние политики запрета ActiveX-объектов на локальном компьютере для всех программ.

Изменение минимального уровня зон Интернета

В браузере Internet Explorer 6.0 реализована защита уровней зон безопасности от неправильного изменения. Например, если вы попытаетесь изменить уровень зоны Интернета на вкладке Безопасность диалога Свойства обозревателя на уровень Ниже среднего, то браузер не разрешит вам это сделать, так как минимальным допустимым уровнем для зоны Интернета является средний уровень. Наложенный запрет для уровней хранится в реестре, с помощью которого можно изменить минимальный допустимый уровень зоны. Это можно сделать, например, для повышения минимального возможного уровня зон. Конечно, можно это сделать и с целью понижения уровня, но понижать минимальные уровни настоятельно не рекомендуется.

Как уже упоминалось при описании трюков с оболочкой браузера, все настройки зон Интернета хранятся в ветви реестра HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Internet Settings\Zones. Она содержит пять разделов, каждый из которых определяет настройки для отдельной зоны Интернета.

Раздел 0 включает в себя настройки для зоны Мой компьютер, то есть настройки безопасности, которые будут применяться при навигации по файловой структуре компьютера с помощью браузера и при просмотре файлов HTML-страниц, хранящихся на вашем жестком диске. С помощью диалога Свойства обозревателя нельзя редактировать зону Мой компьютер, поэтому для нее не существует такого понятия, как минимальный разрешенный уровень зоны.

Остальные разделы определяют зоны, доступ к которым можно получить с помощью вкладки Безопасность диалога Свойства обозревателя. Раздел 1 определяет зону Местная интрасеть, раздел 2 — зону Надежные узлы, раздел 3 — зону Интернет, а раздел 4 — зону Ограниченные узлы. Все эти разделы содержат (помимо других параметров) два DWORD-параметра: MinLevel и RecommendedLevel. Именно значения этих параметров и указывают минимальный уровень безопасности для конкретной зоны — значение первого параметра определяет минимальный допустимый уровень, а значение второго — рекомендуемый уровень. При этом уровень зоны, меньше которого опускаться нельзя, определяется как наибольший из уровней, указанных в значениях этих двух параметров.

Параметры MinLevel и RecommendedLevel должны содержать идентификатор зоны — число, которое определяет данный уровень зоны для браузера. Возможные идентификаторы и соответствующие им уровни также хранятся в реестре — в ветви HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\TemplatePolicies. Ветвь включает в себя четыре раздела: High, Low, Medium и MedLow. Каждый из них определяет настройки безопасности для конкретного уровня зоны (соответственно, для уровней Высокий, Низкий, Средний, Ниже среднего). Кроме настроек безопасности, эти разделы хранят еще и DWORD-параметр TemplateIndex, значение которого определяет идентификатор

для данного уровня безопасности зоны. Например, на моем компьютере данный параметр содержит следующие значения уровней для конкретной зоны:

- `0õ00012000` — Высокий;
- `0õ00010000` — Низкий;
- `0õ00011000` — Средний;
- `0õ00010500` — Ниже среднего.

Теперь попробуем изменить минимальный возможный уровень для конкретной зоны Интернета. Например, для зоны Надежные узлы. По умолчанию минимальным возможным уровнем для данной зоны является уровень Низкий, то есть значения параметров `MinLevel` и `RecommendedLevel` равны `0õ00010000`. Но если изменить значение двух этих параметров, например, на `0õ00010500`, то пользователю будет запрещено устанавливать уровень Низкий для зоны Надежные узлы — минимальным возможным будет уровень Ниже среднего.

Другие параметры реестра

В конце рассказа о параметрах реестра для браузера Internet Explorer рассмотрим некоторые параметры ветви реестра `HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Internet Settings`. Как уже было сказано, она содержит конфигурационные настройки браузера для текущего пользователя, некоторые из них нельзя изменить с помощью диалоговых окон браузера Internet Explorer.

- `MaxConnectionsPer1_0Server` и `MaxConnectionsPerServer` — параметры `DWORD`-типа, определяют максимальное количество сеансов между вашим компьютером и Интернетом, которые может поддерживать браузер (первый из этих параметров определяет максимальное количество сеансов для протокола HTTP 1.0, а второй — для остальных протоколов). Каждое открытое окно браузера или скачиваемый файл занимают один сеанс. По умолчанию эти параметры не существуют, и их нужно создать.

  С одной стороны, если вы постоянно подключены к Интернету, то можете вообще установить значения этих параметров равными `1`. В таком случае ни один троянский конь не сможет подключаться к нему. А можно при необходимости увеличить количество сеансов. Это может понадобиться, например, при использовании специальных программ для скачивания файлов из Интернета, которые применяют несколько сеансов, достигая тем самым большей скорости скачивания.

- `ReceiveTimeout` — этот параметр `DWORD`-типа указывает время загрузки страницы Интернета (в миллисекундах), которое будет ожидать браузер перед тем, как прервать загрузку. При этом если время загрузки истекло, а загружаемая страница так и не ответила, то будет выдано сообщение о невозможности отображения страницы. По умолчанию параметр не существует, и его нужно создать.

  Если вы используете очень медленное модемное подключение и довольно часты случаи отображения сообщения о невозможности загрузки страницы, то можно попробовать увеличить значение этого параметра.

ПРИМЕЧАНИЕ

Если проблема не исчезла, то можно поэкспериментировать с DWORD-параметром MaxHttpRedirect, который определяет максимальное количество перенаправлений, используемых при поиске необходимого сайта. По умолчанию данный параметр отсутствует в реестре.

- `KeepAliveTimeout` — параметр `DWORD`-типа, определяет интервал времени в миллисекундах, который браузер будет сохранять неактивное соединение с Интернетом (посредством пакетов активности). По умолчанию параметр не существует, и его нужно создать.

  Пакеты активности подключения по умолчанию не отправляются, что можно видеть по значению `DWORD`-параметра `DisableKeepAlive` из этой же ветви реестра.

- `User Agent` — этот параметр строкового типа идентифицирует версию вашего браузера (строка обозревателя) для сайтов, которым необходима эта информация для решения, какую страницу вам послать. Другими словами, если сайт имеет несколько вариантов форматирования или HTML-кода — каждый вариант для отдельной версии браузера, — то он запрашивает значение данного параметра, чтобы передать наиболее подходящий для браузера HTML-код. Для шестой версии браузера по умолчанию значение этого параметра равно `Mozilla/4.0 (compatible; MSIE 6.0; Win32)`.

ПРИМЕЧАНИЕ

С помощью реестра можно добавить произвольный текст к строке обозревателя. Для этого предназначена ветвь реестра HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\User Agent\Post Platform. Чтобы добавить строку, нужно создать в этой ветви параметр строкового типа со значением, равным IEAK. Название параметра как раз и будет добавляться к строке обозревателя.

- `FromCacheTimeout` — определяет время ожидания считывания из кэша. По умолчанию параметр не существует, и его нужно создать.

- `SocketSendBufferLength` и `SocketReceiveBufferLength` — эти параметры `DWORD`-типа указывают размеры буферов приема и передачи данных для каждого используемого порта. Если при работе с Интернетом часто возникают ошибки приема/передачи или скорость слишком медленная, то можно попробовать поэкспериментировать с данными параметрами `DWORD`-типа. По умолчанию они отсутствуют в реестре.

- `ProxyServer` — параметр строкового типа, определяет адреса прокси-сервера и порты, которые будут использоваться для подключений по различным протоколам Интернета. Например, значение этого параметра, равное `http=10.1.1.2:80;https=10.1.1.1:80;ftp=10.1.2.3:80; gopher=10.1.2.3:80; socks=10.1.1.1:80`, определяет все возможные протоколы подключения. Параметр также редактируется с помощью диалога

Параметры прокси-сервера, который можно отобразить с помощью вкладки Подключения диалогового окна Свойства обозревателя, если нажать кнопку Настройка LAN данного окна и в появившемся диалоге нажать кнопку Дополнительно (чтобы она стала активной, нужно установить флажок Использовать прокси-сервер для подключения LAN).

Outlook Express

Еще одной программой, настройки которой будут рассмотрены, является стандартный почтовый клиент Outlook Express. Как и раздел о браузере Internet Explorer, этот раздел будет содержать сведения только о тех параметрах, доступ к которым нельзя получить с помощью стандартных диалогов почтового клиента.

Но перед тем, как начать описание параметров реестра, стоит несколько слов сказать о ветвях реестра, в которых почтовый клиент хранит свои настройки. Для этого он использует две основные ветви реестра — `HKEY_CURRENT_USER\Software\Microsoft\Outlook Express\5.0` (ветвь находится и в корневом разделе `HKEY_LOCAL_MACHINE`) и `HKEY_CURRENT_USER\Identities\{GUID-íîìåð óäîñòîâåðåíèÿ ïî÷òîâîãî êëèåíòà}\Software\Microsoft\Outlook Express\5.0`. В первой ветви реестра нет ничего необычного, поэтому о ней больше говориться не будет. А вот вторая ветвь реестра может вызвать вопросы. Например, вопрос о том, что же это за GUID-номер удостоверения пользователя. Все дело в том, что один пользователь может иметь сразу несколько удостоверений в почтовом клиенте Outlook Express, настройки которых как раз и определяются в разделах ветви `HKEY_CURRENT_USER\Identities\{GUID-íîìåð óäîñòîâåðåíèÿ ïî÷òîâîãî êëèåíòà}`. Кроме разделов удостоверения пользователя, ветвь реестра `HKEY_CURRENT_USER\Identities` хранит еще и настройки главной идентификационной записи — они расположены в разделе с GUID-номером `{0EB9C6CE-AD1E-49DD-9965-129A078D453E}`.

Чтобы узнать, какому имени пользователя принадлежат настройки из ветви реестра `HKEY_CURRENT_USER\Identities\{GUID-íîìåð óäîñòîâåðåíèÿ ïî÷òîâîãî êëèåíòà}`, необходимо посмотреть на параметр строкового типа `Username` проверяемой ветви реестра. Например, для ветви реестра `HKEY_CURRENT_USER\Identities\{0EB9C6CE-AD1E-49DD-9965-129A078D453E}` значение параметра `Username` будет равно `Ãëàâíàÿ èäåíòèôèêàöèîííàÿ çàïèñü`.

Оболочка

Как обычно, все параметры реестра будут разделены на два типа — те, что изменяют оболочку программы Outlook Express, и те, что определяют конфигурацию Outlook Express. И, конечно же, сначала рассмотрим некоторые параметры, относящиеся к настройке оболочки почтового клиента. Среди них есть как изменяющие вид окна Outlook Express, например изменяющие заголовок, так и скрывающие различные функциональные возможности почтового клиента.

Изменение заголовка Outlook Express

Чтобы изменить заголовок почтового клиента Outlook Express (точнее, добавить к заголовку текст, отображаемый между названием папки, в которой вы сейчас находитесь, и именем пользователя), достаточно воспользоваться параметром строкового типа `WindowTitle` ветви реестра `HKEY_CURRENT_USER\Identities\{GUID-íîìåð ó÷åòíîé çàïèñè ïî÷òîâîãî êëèåíòà}\Software\Microsoft\Outlook Express\5.0`. По умолчанию данный параметр отсутствует в реестре, поэтому его необходимо создать и присвоить ему значение, определяющее заголовок, который вы хотите использовать для данного удостоверения. Например, результат присвоения значения `Óôåë íà îáåä` параметру `WindowTitle` из ветви реестра `HKEY_CURRENT_USER\Identities\{0EB9C6CE-AD1E-49DD-9965-129A078D453E}\Software\Microsoft\Outlook Express\5.0` можно увидеть на рис. 6.6.

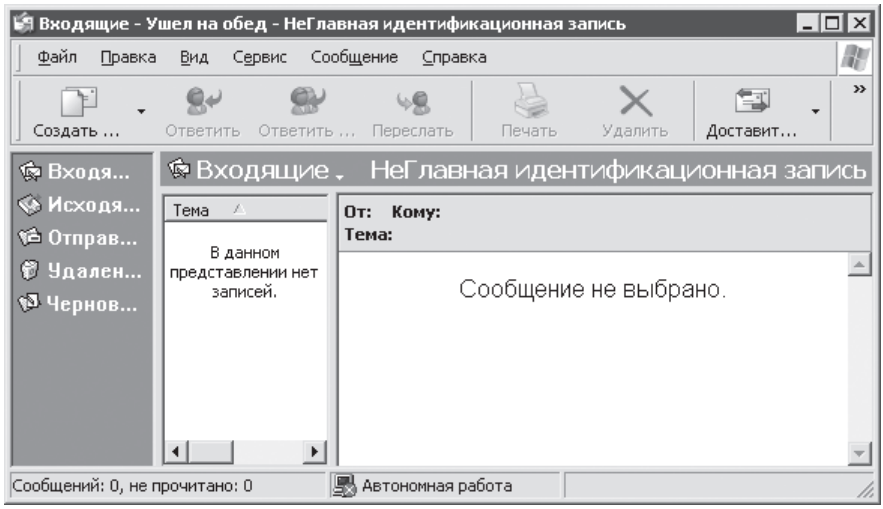


Рис. 6.6. Добавление произвольного текста к заголовку

Запрет отображения заставки Outlook Express

При запуске почтового клиента Outlook Express (после выбора удостоверения и ввода для него пароля, если он необходим, и до непосредственного отображения окна почтового клиента) в течение нескольких секунд отображается заставка, вывод которой можно запретить с помощью параметра реестра. Для этого необходимо в ветви реестра `HKEY_CURRENT_USER\Identities\{0EB9C6CE-AD1E-49DD-9965-129A078D453E}\Software\Microsoft\Outlook Express\5.0` создать `DWORD`-параметр `NoSplash` и присвоить ему значение, равное `1`.

Отображение различных панелей при запуске Outlook Express

По умолчанию Outlook Express при запуске отображает минимальное количество своих панелей. Вкратце опишем параметры реестра, с помощью которых можно

запретить или разрешить отображение панелей при запуске почтового клиента Outlook Express. Все эти параметры имеют тип DWORD и находятся в ветви реестра HKEY_CURRENT_USER\Identities\{GUID-íîìåð óäîñòîâåðåíèÿ ïî÷òîâîãî êëèåíòà}\Software\Microsoft\Outlook Express\5.0.

- Show Outlook Bar — если значение этого параметра равно 1, то слева в окне почтового клиента будет отображаться панель папок Outlook Express (рис. 6.7). По умолчанию данная панель не отображается.

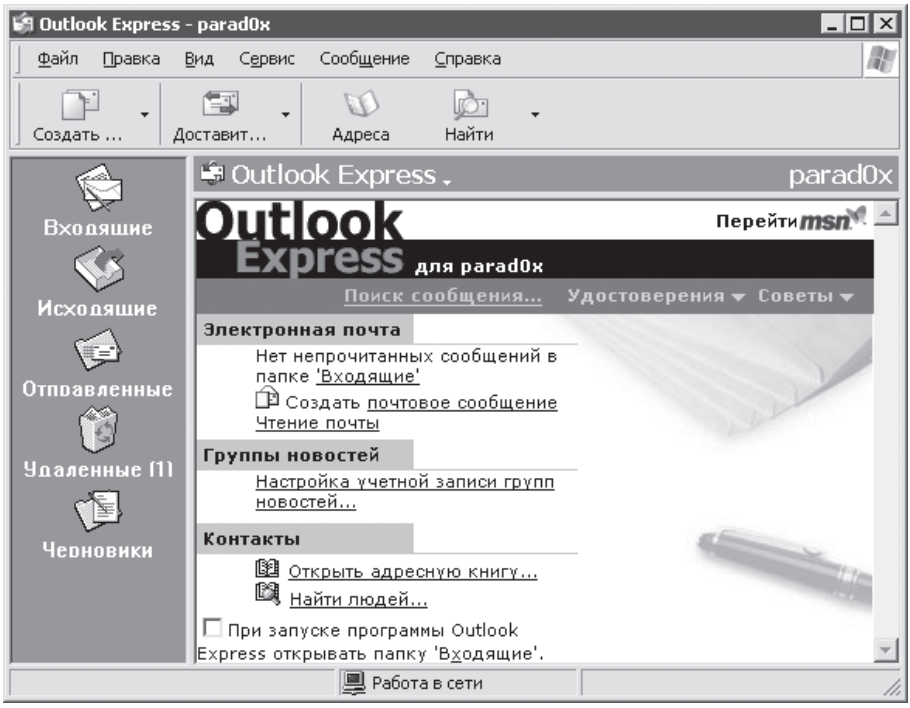


Рис. 6.7. Добавление панели папок

ПРИМЕЧАНИЕ

По умолчанию отображается другая панель папок. Эта панель так и называется — Папки. Ее очень легко скрыть — достаточно нажать крестик в заголовке панели. А вот отобразить эту панель — задача сложнее. Для этого нужно присвоить параметру Tree значение 1. Этот параметр также имеет тип DWORD и находится в ветви системного реестра HKEY_CURRENT_USER\Identities\{GUID-номер удостоверения}\Software\Microsoft\Outlook Express\5.0.

- HideFolderBar — по умолчанию под панелью инструментов находится заголовок, в котором отображено название текущей папки, а также имя удостоверения, используемое в данный момент (см. рис. 6.7). Данный заголовок можно скрыть — для этого достаточно присвоить параметру HideFolderBar значение 1.

- ShowHybridView — если значение этого параметра равно 0, то при просмотре письма не будет отображаться информация заголовка письма (поля От, Кому, Тема). По умолчанию он равен 1 и расположен в ветви системного реестра HKEY_CURRENT_USER\Identities\{GUID-íîìåð óäîñòîâåðåíèÿ}\Software\Microsoft\Outlook Express\5.0\Mail.
- ShowBodyBar — если значение этого параметра равно 1, то при запуске Outlook Express будет отображаться панель тела письма. По умолчанию она не отображается. Есть одна интересная возможность, которую предоставляет панель тела письма, — определение файла рисунка, который будет на ней выводиться. Для этого используется параметр строкового типа BodyBarPath, значение которого как раз и определяет путь к картинке. Параметр расположен в ветви реестра HKEY_CURRENT_USER\Identities\{GUID-íîìåð óäîñòîâåðåíèÿ ïî÷òîâîãî êëèåíòà}\Software\Microsoft\Outlook Express\5.0. На рис. 6.8 отображен пример использования тела письма для отображения рисунка.

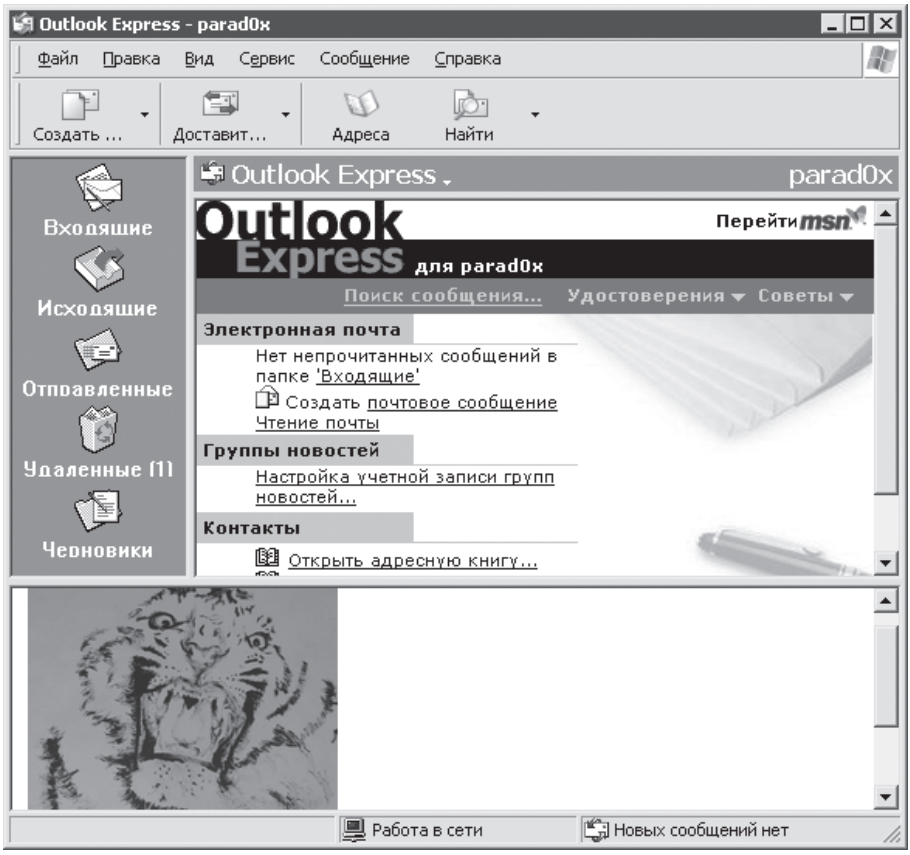


Рис. 6.8. Отображение произвольного рисунка на панели тела письма

- SplitDir — по умолчанию значение параметра равно 0. Это говорит о том, что поля списка сообщений и тела сообщений будут располагаться по горизонтали.

Если же значение равно 1, то эти поля будут отображаться по вертикали (рис. 6.9). Данный параметр расположен в ветви системного реестра HKEY_CURRENT_USER\Identities\{GUID-íîìåð óäîñòîâåðåíèÿ}\Software\Microsoft\Outlook Express\5.0\Mail.

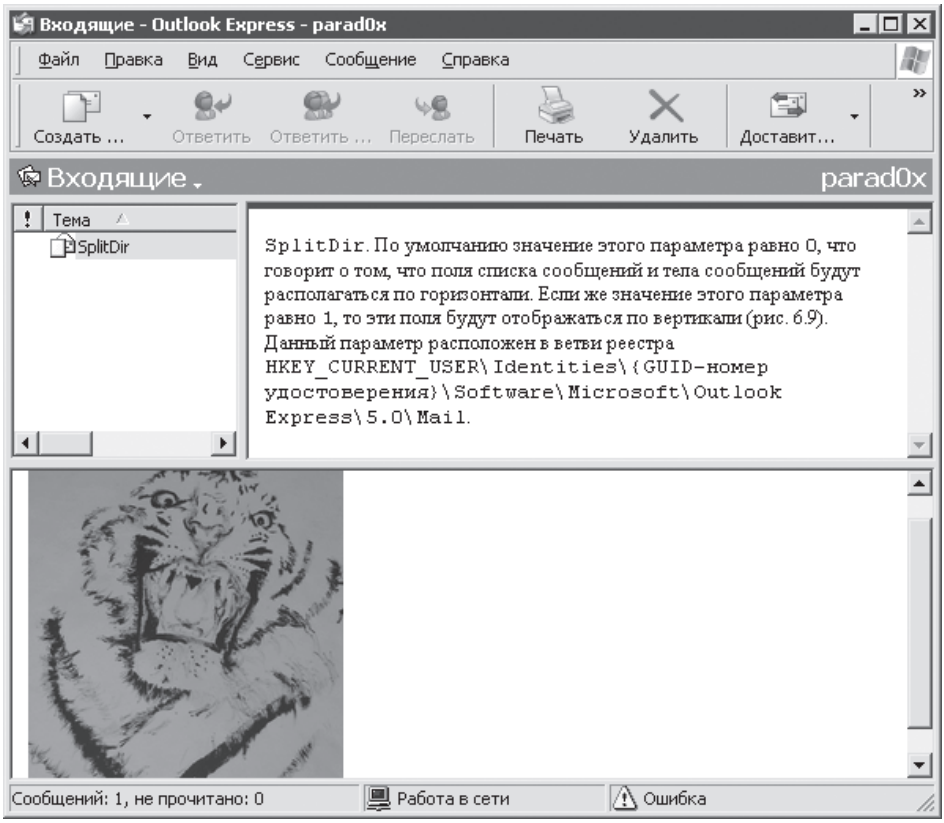


Рис. 6.9. Отображение полей по вертикали

Скрытие различных элементов диалога Параметры

Напоследок будет рассмотрена возможность запрета изменения различных флажков и других элементов диалога Параметры, который можно вызвать с помощью одноименного элемента меню Сервис. Все описываемые параметры имеют тип DWORD.

- RequestMDNLocked — расположен в ветви реестра HKEY_CURRENT_USER\Identities\{GUID-íîìåð óäîñòîâåðåíèÿ}\Software\Microsoft\Outlook Express\5.0. Если его значение равно 1, то будет заблокирован флажок Запрашивать уведомления о прочтении для всех отправляемых сообщений на вкладке Уведомления.
- SendMDNLocked — находится в ветви системного реестра HKEY_CURRENT_USER\Identities\{GUID-íîìåð óäîñòîâåðåíèÿ}\Software\Microsoft\Outlook Express\5.0. Если его значение равно 1, то будут заблокированы

переключатели области Обработка запросов уведомлений о прочтении на вкладке Уведомления.

- `Security Zone Locked` — расположен в ветви реестра `HKEY_CURRENT_USER\Identities\{GUID-íîìåð óäîñòîâåðåíèÿ}\Software\Microsoft\Outlook Express\5.0`. Если его значение равно 1, то будут заблокированы переключатели области Защита от вирусов на вкладке Безопасность.
- `Safe Attachments Locked` — находится в ветви системного реестра Windows `HKEY_CURRENT_USER\Identities\{GUID-íîìåð óäîñòîâåðåíèÿ}\Software\Microsoft\Outlook Express\5.0\Mail`. Если его значение равно 1, то будет заблокирован флажок Не разрешать сохранение или открытие вложений, которые могут содержать вирусы области Защита от вирусов на вкладке Безопасность.
- `Warn on Mapi Send Locked` — расположен в ветви системного реестра Windows `HKEY_CURRENT_USER\Identities\{GUID-íîìåð óäîñòîâåðåíèÿ}\Software\Microsoft\Outlook Express\5.0\Mail`. Если его значение равно 1, то будет заблокирован флажок Предупреждать, если приложения пытаются отправить почту от моего имени области Защита от вирусов на вкладке Безопасность.

По умолчанию этих параметров нет, и их нужно создать.

Для примера на рис. 6.10 приведено отображение вкладки Уведомления диалога Параметры при использовании приведенных выше параметров реестра.

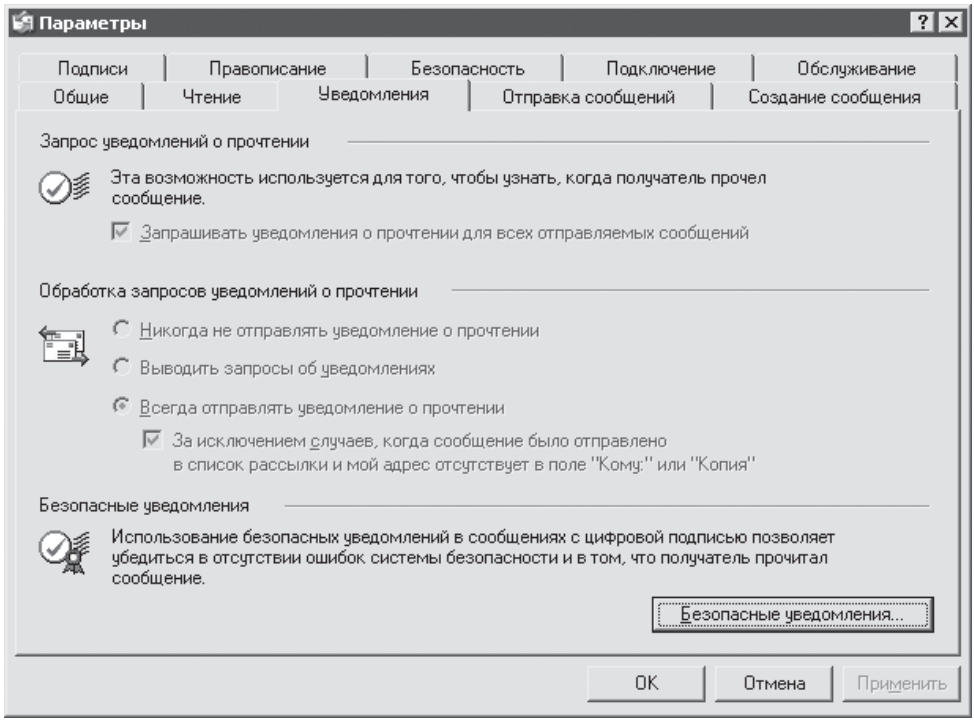


Рис. 6.10. Блокировка некоторых элементов вкладки Уведомления

Конфигурация

Теперь несколько слов скажем о конфигурации Outlook Express. В этом разделе будет рассмотрен способ хранения паролей для удостоверений почтового клиента, а также способ использования этих паролей при запуске почтового клиента Outlook Express.

Пароли Outlook Express

Пароли всех удостоверений Outlook Express хранятся в реестре. Для этого предназначена ветвь реестра `HKEY_CURRENT_USER\Software\Microsoft\Protected Storage System Provider\«èäåíòèôèêàòîð áåçîïàñíîñòè âàøåé ó÷åòíîé çàïèñè (SID)»\Data`. По умолчанию доступ к ветви системного реестра `HKEY_CURRENT_USER\Software\Microsoft\Protected Storage System Provider\«èäåíòèôèêàòîð áåçîïàñíîñòè âàøåé ó÷åòíîé çàïèñè (SID)»` не определен ни для какой учетной записи. Поэтому если ваша учетная запись принадлежит к группе Администраторы, то необходимо будет еще изменить права доступа к этой ветви реестра, чтобы увидеть ее содержимое (учетные записи, принадлежащие к другим группам, не имеют права изменять права доступа к данной ветви реестра). После того как вы измените права доступа, перед вами появятся два дочерних раздела ветви — `Data` и `Data2`. Пароли Outlook Express в зашифрованном виде содержатся в разделе `Data`, поэтому узнать пароль конкретного пользователя не получится, но можно переименовать раздел `Data`, например, в раздел `2Data` или просто удалить его. После этого все пароли на удостоверения данного пользователя будут удалены и можно будет войти в его удостоверения без пароля. Это может понадобиться в том случае, если пароль на удостоверение утерян или поврежден.

ВНИМАНИЕ

Естественно, что администраторы могут удалить пароли удостоверений пользователей не только в том случае, когда пользователи забыли эти пароли, но и в зависимости от своих моральных качеств, просто так, ради интереса. При этом администратор может просто присвоить разделу Data другое имя, а потом, после того как получит доступ к удостоверению, переименовать раздел обратно в Data — и пользователь даже не узнает, что кто-то входил в его удостоверение. Именно поэтому в корпоративной сети следует внимательно относиться к своим удостоверениям, а также к переписке, которая хранится в них.

ПРИМЕЧАНИЕ

Идентификатор безопасности SID является числом в специальном формате (например, для учетной записи Администратор этот идентификатор всегда равен S-1-5-21-1645522239-1957994488-839522115-500). Это число идентифицирует данного пользователя во всех операциях доступа к реестру, файловой системе Windows, сети и т. д. Идентификатор безопасности — это уникальное число (за исключением учетной записи Администратор, для которой идентификатор безопасности на всех компьютерах одинаковый), однозначно идентифицирующее учетную запись. Список всех идентификаторов безопасности для учетных записей (существуют также идентификаторы безопасности для групп пользователей) можно просмотреть в ветви реестра

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\ProfileList. Ветвь содержит разделы, названные в честь идентификаторов безопасности. При этом данные разделы хранят параметр строкового типа ProfileImagePath, который определяет путь к папке профиля, используемой данной учетной записью (конечная папка этого пути названа в честь логина учетной записи, поэтому можно определить, какой учетной записи соответствует каждый идентификатор безопасности).

Но удаление администраторами паролей к удостоверениям пользователей — это еще не все проблемы безопасности в Outlook Express. Существует еще одна. При входе пользователя в почтовый клиент Outlook Express (после ввода пользователем пароля) в ветви реестра `HKEY_CURRENT_USER\Identities` создается `DWORD`-параметр `Identity Login`. Он определяет так называемый идентификатор данного удостоверения (эти идентификаторы статичны, то есть всегда одинаковы для конкретной учетной записи пользователя, а не удостоверения). Идентификатор существует до выхода пользователя из почтового клиента — во время выхода он удаляется. Все дело в том, что если кто-то узнает значение данного идентификатора для удостоверения, например с помощью сетевого доступа просмотрит ветвь реестра `HKEY_CURRENT_USER\Identities` удаленного компьютера или просто посмотрит, чему равен идентификатор при своей работе в почтовом клиенте, то он сможет спокойно зайти в удостоверение, которое открывалось последним — почтовый клиент не потребует ввода пароля. Для этого достаточно будет перед открытием почтового клиента просто создать в ветви системного реестра `HKEY_CURRENT_USER\Identities` параметр `Identity Login` и присвоить ему значение идентификатора.

Теперь рассмотрим небольшой пример. Допустим, существует общественный компьютер, на котором используется почтовый клиент Outlook Express. В этом почтовом клиенте существует три удостоверения, при этом одно из них ваше, например, оно будет иметь GUID-номер удостоверения `{7FA55060-42B6-4CA4-8925-51F7AE55A20F}`, а второе удостоверение, доступ к которому очень нужно получить, имеет GUID-номер `{5E92CB22-3FED-493A-9D6F-F7432CF5CD7C}`. Если вы думаете, что получить доступ к этому удостоверению невозможно, то вы ошибаетесь — на это понадобится пять минут. Попробуем это сделать. Для начала, конечно, нужно зайти в свое удостоверение и посмотреть значение `DWORD`-параметра `Identity Login` ветви реестра `HKEY_CURRENT_USER\Identities`. После того как вы запомнили значение этого параметра, закрывайте почтовый клиент (внимание, нужно просто закрыть почтовый клиент, а не завершить сеанс работы с удостоверением). Вот, в принципе, и все. Теперь осталось выполнить последний шаг — отредактировать три параметра ветви реестра `HKEY_CURRENT_USER\Identities`.

1. `Identity Login` — необходимо создать этот параметр `DWORD`-типа и присвоить ему значение, которое вы недавно запомнили.
2. `Last User ID` — нужно присвоить этому параметру строкового типа значение GUID-номера удостоверения, к которому нужно получить доступ. В вашем случае, нужно присвоить значение `{5E92CB22-3FED-493A-9D6F-F7432CF5CD7C}` (если вы этого не сделаете, то просто войдете без пароля в свое удостоверение).

3. `Last Username` — необходимо присвоить этому параметру строкового типа значение логина удостоверения, доступ к которому нужно получить (еще не забыли, что его можно посмотреть в значении параметра `Username` ветви системного реестра `HKEY_CURRENT_USER\Identities\{GUID-íîìåð óäîñòîâåðåíèÿ ïî÷òîâîãî êëèåíòà, ê êîòîðîìó âû õîòèòå ïîëó÷èòü äîñòóï}`?).

Вот теперь точно все — достаточно еще раз открыть почтовый клиент, и вы без знания пароля войдете в удостоверение пользователя, имеющее GUID-номер `{5E92CB22-3FED-493A-9D6F-F7432CF5CD7C}`.

Как видите, проблема удаления паролей администраторами компьютера еще не самая страшная из всех, ведь предыдущий трюк может сделать обычный пользователь. Поэтому если корреспонденция, которую вы отправляете, имеет какую-то ценность, то ни в коем случае нельзя пользоваться для работы с ней почтовым клиентом Outlook Express, да еще и на общественных компьютерах. Ведь каким бы длинным и сложным ни был ваш пароль, его всегда можно удалить или обойти за пять минут.

Скрытие возможности редактирования списка учетных записей Outlook Express

Учетная запись — это сведения о пароле, логине, сайте почтового сервера и настройках для подключения к этому сайту сервера, с которого будет скачиваться ваша почта, если это почтовый сервер, или новости, если новостной. Каждое удостоверение содержит как общие учетные записи, создаваемые при установке почтового клиента, так и уникальные, создаваемые пользователем, которому принадлежит удостоверение. В некоторых случаях нет необходимости разрешать пользователям создавать свои учетные записи, например, когда почтовый клиент должен использоваться только для отправки почты на какой-то стандартный почтовый сервер. В этом случае можно скрыть команду `Учетные записи` из меню `Сервис` почтового клиента Outlook Express. Для этого достаточно воспользоваться `DWORD`-параметром `No Modify Accts` из ветви реестра `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Outlook Express`. По умолчанию этого параметра нет, и его нужно создать. Необходимо присвоить ему значение `1`.

При использовании этого параметра стоит иметь в виду, что добавить учетную запись или отредактировать свойства уже существующей все-таки можно — для этого достаточно воспользоваться реестром. Сведения об учетных записях, принадлежащих данному удостоверению, находятся в ветви реестра `HKEY_CURRENT_USER\Identities\{GUID-íîìåð óäîñòîâåðåíèÿ}\Software\Microsoft\Internet Account Manager\Accounts`, полный доступ к содержимому которой имеют все пользователи. Эта ветвь не существует для главной учетной записи, для созданных самостоятельно должна существовать. Она включает в себя список разделов, каждый из них определяет настройки одной учетной записи данного удостоверения, и чтобы добавить свою учетную запись, нужно создать новый раздел. Например, в командной строке можно воспользоваться следующей командой: `reg copy «HKEY_CURRENT_USER\Identities\{GUID-íîìåð óäîñòîâåðåíèÿ}\Software\Microsoft\Internet Account Manager\Accounts\«íàçâàíèå`

ñóùåñòâóþùåãî ðàçäåëà»» «HKEY_CURRENT_USER\Identities\{GUID-íîìåð óäîñòîâåðåíèÿ}\Software\Microsoft\Internet Account Manager\Accounts\«íàçâàíèå íîâîãî ðàçäåëà»» /s. После выполнения этой команды в ветви HKEY_CURRENT_USER\Identities\{GUID-íîìåð óäîñòîâåðåíèÿ}\Software\Microsoft\Internet Account Manager\Accounts будет создан новый раздел, содержимое которого придется самостоятельно изменить.

После того как будет настроена новая учетная запись, необходимо будет еще сделать ее учетной записью для работы с почтой по умолчанию. Для этого используется строковый параметр Default Mail Account (для почтового сервера) или строковый параметр Default LDAP Account (для Active Directory). Оба этих параметра расположены в ветви реестра HKEY_CURRENT_USER\Identities\{GUID-íîìåð óäîñòîâåðåíèÿ}\Software\Microsoft\Internet Account Manager и хранят название раздела реестра, содержащего настройки соответствующей учетной записи (название раздела, который был создан на предыдущем этапе).

Как видите, запрет, накладываемый параметром No Modify Accts, также очень легко обойти, поэтому наиболее действенным запретом на редактирование учетных записей по-прежнему остается изменение прав доступа к соответствующим ветвям реестра.

Скрытие возможности работы с Windows Messenger

Существует возможность скрытия команд Состояние и Windows Messenger из меню Сервис. Для этого достаточно присвоить DWORD-параметру Hide Messenger значение, равное 2. Этот параметр расположен в ветви реестра HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Outlook Express. По умолчанию параметр не существует, и его нужно создать.

Глава 7

Оптимизация Windows

- **Компоненты Windows XP**
- **Сеть и сетевые компоненты**
- **Другие способы оптимизации Windows**

В данной главе речь пойдет о параметрах реестра, с помощью которых можно повысить скорость или качество работы различных компонентов Windows XP, например таких, как сетевое соединение и стеки протоколов, а также скорость самой операционной системы. Следует сказать, что в этой главе не будет приведено никаких советов или правил изменения тех или иных параметров реестра. Здесь автор хотел бы предложить набор интересных параметров реестра, направленных на оптимизацию компонентов Windows, а вы уж сами решите, какие параметры стоит изменить, а какие лучше не трогать.

Компоненты Windows XP

Windows XP по сравнению с предыдущими версиями Windows сделала большой шаг в повышении оптимизации работы операционной системы. Это выразилось в добавлении новых программ и служб, направленных на автоматическую оптимизацию операционной системы, а также в появлении новых параметров реестра. Но обо всем по порядку.

Службы

Службами являются специальные программы, с помощью которых реализуются те или иные функциональные возможности Windows. Службы, как правило, не имеют пользовательского интерфейса и выполняются в фоновом режиме. В этом они практически ничем не отличаются от программ-демонов в UNIX-совместимых системах или резидентных программ в MS-DOS. Конечно, службы являются очень полезными программами, с помощью которых повышается функциональная составляющая Windows, и тем не менее не всегда функции, реализуемые той или иной службой, могут быть необходимы на компьютере. Поэтому некоторые из них рекомендуется отключать, ведь каждая работающая служба занимает некоторый объем оперативной памяти, а иногда и некоторую часть производительности процессора. Но перед отключением служб рассмотрим те функции, которые они предоставляют.

DHCP-клиент

Данная служба используется при существовании в сети DHCP-сервера. DHCP-сервер предназначен для выдачи всем компьютерам, не имеющим постоянного IP-адреса, временного IP-адреса, чтобы они могли работать в сети. Служба является отличным средством автоматизирования процесса настройки стека TCP/IP на всех компьютерах сети. Если раньше администратору приходилось самому настраивать каждый компьютер сети, то теперь с помощью DHCP-сервера он может просто определить настройки параметров стека TCP/IP и пул IP-адресов, выдаваемых каждому компьютеру, который будет просить об этом DHCP-сервер. При этом просьба осуществляется при помощи службы `DHCP-клиент`. При входе компьютера в сеть служба отправляет широковещательный запрос с просьбой выдачи временного IP-адреса всем DHCP-серверам, которые присутствуют в сети. После этого каждый DHCP-сервер просматривает свой пул адресов, чтобы определить, может ли он предоставить требуемый IP-адрес. В случае, если пул DHCP-сервера

имеет IP-адрес, удовлетворяющий сети, в которой находится DHCP-клиент, он высылает запрос на предложение данного IP-адреса. После того, как DHCP-клиент получает запрос на предложение IP-адреса, он опять отправляет широковещательный запрос всем DHCP-серверам. На этот раз в широковещательном запросе говорится, что DHCP-клиент принимает IP-адрес. Во-первых, с помощью этого широковещательного запроса DHCP-сервер уведомляется, чей IP-адрес принят, а во-вторых, все остальные DHCP-серверы, которые также выслали запросы с предложениями IP-адреса для данного DHCP-клиента, понимают, что DHCP-клиент выбрал не их адрес и что их услуги больше не нужны. После того как DHCP-сервер, чье предложение IP-адреса было выбрано, получает широковещательный запрос, он высылает подтверждение на передачу IP-адреса DHCP-клиенту (и информацию о сроке аренды IP-адреса), а также дополнительные настройки стека TCP/IP (их называют параметрами), которые необходимо использовать для работы в сети.

Служба DHCP-клиент использует около 20 Кбайт памяти и запускается с правами локальной системы (Local System) автоматически при каждом входе пользователя в систему (при этом она запускается как часть процесса `svchost.exe`).

ПРИМЕЧАНИЕ

Как правило, существует три основных учетных записи, с правами которых запускаются службы. Это учетная запись локальной системы (Local System), локальной службы (NT AUTHORITY\LocalService), а также учетная запись сетевой службы (NT AUTHORITY\NetworkService). Учетная запись локальной системы обладает максимальными правами на доступ к компонентам операционной системы (создание учетных записей администраторов, редактирование любых ветвей реестра, а также доступ к любым файлам и папкам файловой системы). Учетные записи локальной и сетевой служб обладают всеми правами, доступными группе Пользователи.

Как уже говорилось выше, данная служба необходима только в том случае, если существует сеть и в этой сети есть хотя бы один DHCP-сервер. Согласитесь, не многим домашним компьютерам может понадобиться служба DHCP-клиент, поэтому ее запуск можно отключить. Для этого необходимо DWORD-параметру `Start`, расположенному в ветви системного реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Dhcp`, присвоить значение `4`.

ПРИМЕЧАНИЕ

Не обязательно для отключения служб пользоваться реестром — намного легче это сделать с помощью консоли services.msc, речь о которой пойдет в следующей главе книги. Для остановки/запуска службы можно воспользоваться следующими командами программы net: Net start — запустить службу; Net stop — остановить службу; Net pause — приостановить службу; Net continue — продолжить работу приостановленной службы.

В качестве параметра запуска все эти команды требуют для своей работы указания имени службы (команду net start можно запускать и без указания запускаемой службы, в этом случае она отобразит список всех запущенных в данный момент служб).

Имя службы является названием раздела реестра, в котором содержатся параметры настройки службы. Например, для службы DHCP-клиент раздел реестра называется Dhcp (другими словами, чтобы запустить службу DHCP-клиент, нужно воспользоваться командой net start dhcp). А чтобы запустить рассматриваемую далее службу DNS-клиент, которая хранит свои настройки в разделе реестра DNSCACHE, нужно воспользоваться командой net start dnscache.

Для запуска службы DHCP-клиент нужно, чтобы также были запущены следующие службы: Драйвер протокола TCP/IP (для его работы необходима служба Драйвер IPSEC), AFD и NetBios через TCP/IP (для его работы требуется служба Драйвер протокола TCP/IP). Для запуска этих служб также необходимо воспользоваться DWORD-параметром Start, которому нужно присвоить значение 0 (запускать службу при загрузке системы), 1 (запускать службу при инициализации операционной системы), 2 (запускать службу автоматически) или 3 (запускать службу вручную). Этот параметр должен находиться в следующих ветвях реестра:

- HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\AFD — для службы AFD (параметр Start должен иметь значение 1);
- HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip — для службы Драйвер протокола TCP/IP (параметр Start должен иметь значение 1);
- HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\IPSec — для службы Драйвер IPSEC (параметр Start должен иметь значение 1);
- HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\NetBT — для службы NetBios через TCP/IP (параметр Start должен иметь значение 1).

Для работы службы DHCP-клиент также необходима библиотека dhcpcsvc.dll.

ПРИМЕЧАНИЕ

Как правило, сведения о необходимой для работы службы библиотеки находятся в параметре строкового типа ServiceDll, расположенном в разделе Parameters ветви реестра, содержащей настройки службы. В этом разделе также может храниться DWORD-параметр ServiceDllUnloadOnStop. Если его значение равно 1, то при остановке службы из памяти будет выгружаться библиотека, необходимая для ее работы.

DNS-клиент

Служба предназначена для получения IP-адреса удаленного компьютера при известном доменном или url-адресе этого компьютера (например, www.mail.ru). При этом процесс получения IP-адреса удаленного компьютера реализуется благодаря взаимодействию службы DNS-клиент с DNS-сервером. Это взаимодействие начинается после ввода запроса на подключение к удаленному компьютеру с использованием доменного имени компьютера (например, при вводе в адресную строку браузера адреса www.mail.ru). После этого служба DNS-клиент пытается найти IP-адрес компьютера, соответствующий введенному доменному или url-адресу, в своем кэше (данный кэш существует до окончания работы службы DNS-клиент и хранит соответствия всех IP-адресов доменным именам, которые уже были

найдены службой DNS-сервер). Если служба DNS-клиент не находит в кэше соответствующий доменному имени IP-адрес, она обращается к содержимому файла HOSTS (если, конечно, обращение к данному файлу разрешено), расположенному на локальном компьютере (в каталоге %SystemRoot%\System32\drivers\etc) и включающему в себя соответствия между доменными именами и IP-адресами компьютеров, которым эти имена принадлежат. Если же и в этом файле нет сведений об IP-адресе необходимого компьютера, то служба обращается к DNS-серверу, используемому для разрешения имен компьютеров по умолчанию (в сети может существовать несколько DNS-серверов, при этом один из них является основным, к которому и обращаются компьютеры для разрешения имен). DNS-клиент ищет сведения об IP-адресе компьютера, которому принадлежит данное доменное имя, в своей базе данных. Если в базе данных DNS-сервера нет сведений о соответствующем этому доменному имени IP-адресе, то DNS-сервер просматривает свой кэш уже разрешенных имен компьютеров. Если и кэш не содержит необходимого IP-адреса, то DNS-сервер обращается с запросом на разрешение имени к вышестоящему DNS-серверу (например, если данный DNS-сервер включает в себя сведения о домене narod.ru, то DNS-сервер обращается к вышестоящему DNS-серверу, содержащему сведения о домене ru и т. д.). В итоге, если разрешение IP-адреса все-таки удалось, то IP-адрес, соответствующий данному доменному имени, передается DNS-клиенту, который, в свою очередь, передает его программе, запросившей у него разрешение имени (не забыв перед этим поместить данное разрешение имен в свой кэш). Если же разрешение имени не удалось, то программа оповестит об этом пользователя, сказав ему, что компьютер с введенным именем не найден.

ПРИМЕЧАНИЕ

Как уже говорилось, файл hosts расположен в каталоге %systemroot%\system\drivers\ets и используется в случае, если в кэше DNS-клиента нет сведений о разрешении данного доменного или url-имени. Файл hosts является обычным текстовым файлом, содержащим соответствия IP-адреса компьютера его url-адресу. Вы сами можете создать данные соответствия для часто открываемых в Интернете сайтов, чтобы они открывались быстрее и при открытии загружали меньше трафика (ведь браузеру не придется обращаться к DNS-серверу). Для этого достаточно в файле hosts создать строку такого вида: IP-адрес URL-адрес. Например, можно разрешить IP-адрес сайта www.mail.ru. Его url-имя у вас есть (www.mail.ru), но как узнать IP-адрес? Для этого вам понадобится программа командной строки ping.exe. Необходимо запустить командную строку и ввести команду ping www.mail.ru, после чего программа выведет IP-адрес, принадлежащий url-имени www.mail.ru. Для www.mail.ru это будет адрес 194.67.57.26, то есть в файле hosts нужно создать строку вида 194.67.57.26 www.mail.ru.

С помощью файла hosts можно также бороться с баннерными серверами. Для этого достаточно разрешить имя сайта, который раздает другим сайтам баннеры, на IP-адрес своего компьютера (например, с помощью строки 127.0.0.1 www.banners.com), и баннеры от этого сайта больше не будут загружаться.

Использование файла hosts может быть полезно на домашних компьютерах, подключенных к Интернету, но в коммерческих сетях его использование не рекомендуется.

Служба DNS-клиент занимает около 2 604 Кбайт памяти и запускается с правами сетевой службы (NT AUTHORITY\NetworkService) автоматически при каждом входе пользователя в систему (при этом она запускается как отдельный процесс `svchost.exe`). Данная служба необходима, если в сети присутствует DNS-сервер или компьютер принадлежит к Active Directory (Active Directory уже предполагает, что в сети есть DNS-сервер, ведь без него Active Directory нельзя будет установить). Если эти условия не выполняются, то службу DNS-клиент можно отключить (можно подумать, что эта служба также необходима для подключения к Интернету, но, как показали исследования, это не так, хотя без ее использования качество поиска страниц в Интернете может пострадать, поэтому не рекомендуется отключать данную службу, если вы подключены к Интернету). Для этого необходимо DWORD-параметру `Start`, расположенному в ветви реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Dnscache`, присвоить значение 4.

Для запуска службы DNS-клиент необходимо, чтобы уже была запущена служба Драйвер протокола TCP/IP.

Для работы службы DNS-клиент также необходима библиотека `dnsrslvr.dll`.

Plug and Play

Служба реализует возможность автоматического распознавания и установки новых устройств формата Plug and Play. При этом для пользователя процесс установки происходит прозрачно (или с минимальным участием пользователя). Например, при подключении USB-устройства от пользователя не требуется никаких действий — служба Plug and Play сама найдет драйверы, необходимые для данного устройства, выберет из найденных драйверов тот, который более всего подходит для устройства, и установит его, после чего настроит параметры шины для работы с новым устройством.

Служба Plug and Play занимает около 100 Кбайт оперативной памяти и запускается с правами локальной системы (Local System) автоматически при каждом входе пользователя в систему (при этом она запускается как часть процесса `services.exe`). Отключение данной службы крайне нежелательно, так как после этого система может работать нестабильно, поэтому эта служба должна всегда запускаться автоматически (параметр `Start`, расположенный в ветви реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\PlugPlay`, должен быть равен 2).

На возможность запуска данной службы не влияют никакие другие службы.

Windows Audio

Служба управляет возможностью воспроизведения звука и звуковых эффектов в Windows-программах. В частности, данная служба необходима для работы программы Windows Media Player, поэтому если она будет отключена, то проигрыватель при попытке воспроизведения файла будет выдавать сообщения о том, что звуковая плата отсутствует или для нее не установлены драйверы.

Служба Windows Audio занимает около 300 Кбайт оперативной памяти и запускается с правами локальной системы (Local System) автоматически при каждом входе

пользователя в систему (при этом она запускается как часть процесса `svchost.exe`). Если на компьютере отсутствует звуковая плата или возможность воспроизведения компьютером звуков не используется, то можно отключить данную службу. Иначе лучше ее не трогать. Для отключения службы необходимо DWORD-параметру `Start`, расположенному в ветви реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\AudioSrv`, присвоить значение 4.

Для запуска службы Windows Audio необходимо, чтобы были также запущены службы Plug and Play и Удаленный вызов процедур (RPC). Расположение в реестре службы Plug and Play уже рассматривалось, а служба Удаленный вызов процедур (RPC) хранит свои параметры в ветви реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\RpcSs`.

Для работы службы Windows Audio необходима библиотека `audiosrv.dll`.

Автоматическое обновление

Служба предназначена для автоматического скачивания из Интернета и установки обновлений операционной системы и стандартных компонентов Windows XP. При этом сведения об уже установленных обновлениях берутся из реестра. Для этого предназначены две ветви реестра — `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Hotfix` и `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Updates`. Первая содержит список разделов, названных в честь установленных обновлений. Из этих разделов можно определить описание обновления (параметр строкового типа `Fix Description`), а также работоспособность обновления (DWORD-параметр `Valid`). Вторая ветвь реестра также включает в себя разделы с именами, соответствующими установленным обновлениям, но эти разделы содержат намного больше параметров. В частности, в данных разделах находятся сведения о дате установки обновления (параметр строкового типа `InstalledDate`), учетной записи пользователя, который устанавливал данное обновление (параметр строкового типа `InstalledBy`), а также команда, с помощью которой можно удалить установленное обновление, вернув систему к предыдущему состоянию (параметр строкового типа `UninstallCommand`). В этих разделах также находится вложенный раздел `Filelist`, определяющий те системные файлы Windows XP, которые были заменены данным обновлением.

Служба Автоматическое обновление занимает около 800 Кбайт оперативной памяти и запускается с правами локальной системы (Local System) автоматически при каждом входе пользователя в систему (при этом она запускается как часть процесса `svchost.exe`). Эта служба является хорошим средством обновления операционной системы, но только в том случае, если компьютер подключен к Интернету и у вас много денег, чтобы платить за те мегабайты обновлений, которые для вас будет выкачивать автоматическое обновление. В противном случае службу Автоматическое обновление лучше отключить. Для этого достаточно присвоить DWORD-параметру `Start` значение, равное 4. Параметр расположен в ветви реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\wuauserv`.

На возможность запуска данной службы не влияют никакие другие службы. При этом для работы автоматического обновления необходима библиотека `wuauserv.dll`.

Беспроводная настройка

Служба предназначена для автоматической настройки устройств, работающих по стандарту 802.11 (другими словами, использующих беспроводной доступ к сети или компьютеру). При этом многие такие устройства поставляются с собственными программами для установки, настройки и поддержки в функциональном состоянии оборудования, работающего в стандарте 802.11. В этом случае для стабильной работы устройств необходимо использовать только программы, поставляемые вместе с самим устройством, а не средства операционной системы Windows.

Служба Беспроводная настройка занимает около 270 Кбайт оперативной памяти и запускается с правами локальной системы (Local System) автоматически при каждом входе пользователя в систему (при этом она запускается как часть процесса `svchost.exe`). Если на компьютере отсутствуют устройства, работающие в стандарте 802.11, то эту службу можно отключить. Ее также можно отключить, если для настройки и установки устройства предназначена отдельная программа, поставляемая вместе с устройством. Но в этом случае стоит внимательно проследить за тем, не нарушилась ли после отключения службы какая-то функциональная составляющая устройства, и при неправильной работе каких-нибудь функций, опять установить автоматический запуск службы Беспроводная настройка. Чтобы отключить эту службу, достаточно присвоить `DWORD`-параметру `Start` значение, равное 4. Параметр расположен в ветви реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\WZCSVC`.

Для запуска службы Беспроводная настройка необходимо, чтобы были запущены службы NDIS-протокол ввода/вывода пользовательского режима и Удаленный вызов процедур (RPC). При этом первая служба описывается в ветви реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Ndisuio`, а ветвь реестра для второй службы уже рассматривалась.

Для работы службы Беспроводная настройка необходима библиотека `wzcsvc.dll`.

Брандмауэр Windows/Общий доступ к Интернету (ICS)

Служба управляет стандартным брандмауэром Windows, а также возможностью общего доступа к Интернету (ICS). Стандартный брандмауэр Windows предоставляет минимальные функции обеспечения безопасности подключения к Интернету (он следит за всеми открытыми портами и извещает пользователя о попытке какой-либо программы передать данные из Интернета или в Интернет по одному из портов). С помощью ICS сетевые компьютеры, не имеющие модема или подключения к Интернету, могут подключиться к нему по сети, используя подключение ICS-сервера (компьютера с подключением к Интернету и настроенным ICS-сервером).

Служба Брандмауэр Windows/Общий доступ к Интернету (ICS) занимает около 4360 Кбайт оперативной памяти и запускается с правами локальной системы (Local System)

автоматически при каждом входе пользователя в систему (при этом она запускается как часть процесса `svchost.exe`). Если вы используете брандмауэр сторонней фирмы и при этом не применяете функцию ICS, то данную службу можно отключить. Для этого необходимо воспользоваться параметром `Start` из ветви реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\SharedAccess`.

Для запуска службы Брандмауэр Windows/Общий доступ к Интернету (ICS) необходимо, чтобы были запущены следующие службы: Сетевые подключения и Инструментарий управления Windows. Первая служба описывается в ветви системного реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Netman`, а вторая — в ветви реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\winmgmt`.

Для работы службы Брандмауэр Windows/Общий доступ к Интернету (ICS) необходима библиотека `ipnathlp.dll`.

Веб-клиент

Служба позволяет изменять или добавлять файлы, хранящиеся в Интернете. Если эта стандартная функция Windows вам не нужна, то службу лучше отключить.

Служба Веб-клиент занимает около 800 Кбайт оперативной памяти и запускается с правами локальной службы (NT AUTHORITY\LocalService) автоматически при каждом входе пользователя в систему (при этом она запускается как часть процесса `svchost.exe`). Чтобы отключить эту службу, необходимо воспользоваться параметром `Start` из ветви реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\WebClient`.

Для запуска службы Веб-клиент необходимо, чтобы была запущена служба Перенаправить клиентов WebDav. Она описывается в ветви реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\MRxDAV`.

Для работы службы Веб-клиент необходима библиотека `webclnt.dll`.

Вторичный вход в систему

Служба реализует возможность запуска программ от имени другого пользователя. Если она будет остановлена, то вы не сможете воспользоваться командой `runas` (формат запуска программы от имени другого пользователя таков: `runas /user:«ïîëüçîâàòåëü» «ïðîãðàììà»`, например, `runas /user:àäìèíèñòðàòîð mmc.exe`) для запуска программ или оснасток консоли с правами администратора или другого локального пользователя, потому что при запуске будет возникать ошибка. Если вы не пользуетесь данной возможностью, то можете отключить эту службу.

Служба Вторичный вход в систему занимает около 40 Кбайт оперативной памяти и запускается с правами локальной системы (Local System) автоматически при каждом входе пользователя в систему (при этом она запускается как часть процес-

са svchost.exe). Чтобы отключить эту службу, необходимо воспользоваться параметром Start из ветви системного реестра HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\seclogon.

На возможность запуска этой службы не влияют никакие другие службы. Но для ее работы необходима библиотека seclogon.dll.

Диспетчер логических дисков

Служба предназначена для обнаружения и наблюдения за работой новых жестких дисков. При этом все собираемые сведения передаются службе управления диспетчера логических дисков. Иными словами, если служба Диспетчер логических дисков остановлена, то не сможет работать и Служба администрирования диспетчера логических дисков (она запускается вручную программами, которым необходимо получить сведения о конфигурации жестких дисков, и занимает в оперативной памяти около 2700 Кбайт).

Служба Диспетчер логических дисков занимает около 20 Кбайт оперативной памяти и запускается с правами локальной системы (Local System) автоматически при каждом входе пользователя в систему (при этом она запускается как часть процесса svchost.exe). Чтобы отключить эту службу, необходимо воспользоваться параметром Start из ветви реестра HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\dmserver.

Для запуска службы Диспетчер логических дисков необходимо, чтобы были запущены службы Plug and Play и Удаленный вызов процедур (RPC). Ветви реестра, используемые этими службами, уже рассматривались ранее.

Для того чтобы работала служба Диспетчер логических дисков, необходима библиотека dmserver.dll.

Диспетчер очереди печати

Служба предназначена для помещения в оперативную память документов, которые стоят в очереди на печать. Другими словами, эта служба принимает данные, отсылаемые пользователем на печать, и передает их доступному принтеру. Если на компьютере отсутствуют установленные принтеры, то службу можно отключить.

ПРИМЕЧАНИЕ

Если принтер при попытке печати документов не отвечает, попробуйте остановить и запустить данную службу. В некоторых случаях это помогает. Правда, после этого очередь ожидающих печати документов очищается.

Служба Диспетчер очереди печати занимает около 4600 Кбайт оперативной памяти и запускается с правами локальной системы (Local System) автоматически при каждом входе пользователя в систему (как процесс spoolsv.exe). Чтобы отключить

эту службу, необходимо воспользоваться параметром `Start` из ветви реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Spooler`.

Для запуска службы Диспетчер очереди печати необходимо, чтобы была запущена служба Удаленный вызов процедур (RPC).

Диспетчер учетных записей безопасности

Служба обеспечивает безопасность учетной записи локального пользователя. Отключение данной службы, как правило, не влияет на стабильность работы компьютера, но, тем не менее, крайне не рекомендуется и может понизить уровень защищенности операционной системы.

Служба Диспетчер учетных записей безопасности занимает около 8 Кбайт оперативной памяти и запускается с правами локальной системы (Local System) автоматически при каждом входе пользователя в систему (запускается как часть процесса `lsass.exe`). Чтобы отключить эту службу, необходимо воспользоваться параметром `Start` из ветви реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\SamSs`.

Для запуска службы Диспетчер учетных записей безопасности необходимо, чтобы была запущена служба Удаленный вызов процедур (RPC).

Журнал событий

Служба обеспечивает запись сообщений в стандартные журналы Windows (Система, Приложения, Безопасность), просмотреть которые можно при помощи оснастки `eventvwr.msc`. Далее в этой книге оснастка Просмотр событий (`eventvwr.msc`) будет описана подробней. Служба Журнал событий является критически важной службой для работы операционной системы, и уже работающую службу остановить нельзя. Кроме того, если автоматический запуск данной службы по каким-либо причинам будет неудачным, то компьютер начнет перезагрузку. Тем не менее после отключения данной службы компьютер будет работать стабильно, но запуск операционной системы может замедлиться в несколько раз, а журнал Просмотр событий будет пуст. Именно поэтому крайне не рекомендуется отключать эту службу.

Служба Журнал событий занимает около 200 Кбайт оперативной памяти и запускается с правами локальной системы (Local System) автоматически при каждом входе пользователя в систему (запускается как часть процесса `services.exe`). Чтобы отключить эту службу, необходимо воспользоваться параметром `Start` из ветви реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Eventlog`.

На возможность запуска данной службы не влияют никакие другие службы. Но для работы службы Журнал событий необходима библиотека `els.dll`.

Защищенное хранилище

Служба является посредником между частью оперативной памяти, содержащей пароли пользователей, сеансовые билеты и другую критически важную для безопасности компьютера информацию (информация хранится в виде хэша), и про-

граммами, которым нужна для работы информация из защищенного хранилища. Крайне не рекомендуется отключение данной службы, так как это резко снижает уровень защищенности операционной системы.

Служба Защищенное хранилище занимает около 460 Кбайт оперативной памяти и запускается с правами локальной системы (Local System) автоматически при каждом входе пользователя в систему (запускается как часть процесса `lsass.exe`). Чтобы отключить эту службу, необходимо воспользоваться параметром `Start` из ветви реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\ProtectedStorage`.

Для запуска службы Защищенное хранилище необходимо, чтобы была запущена служба Удаленный вызов процедур (RPC).

Инструментарий управления Windows

Служба предоставляет информацию о конфигурации компьютера, установленных на нем программ и служб всем программам, которым она необходима для корректной работы. После отключения данной службы программы, которым необходима информация, предоставляемая службой, могут работать некорректно (кроме того, после ее отключения может некорректно работать служба Запуск серверных процессов DCOM). После отключения данной службы будут отключены службы Центр обеспечения безопасности и Брандмауэр Windows/Общий доступ к Интернету (ICS). Крайне не рекомендуется отключать эту службу.

Служба Инструментарий управления Windows запускается с правами локальной системы (Local System) автоматически при каждом входе пользователя в систему (запускается как часть процесса `svchost.exe`). Чтобы отключить эту службу, необходимо воспользоваться параметром `Start` из ветви реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\winmgmt`.

Для запуска службы Инструментарий управления Windows необходимо, чтобы была запущена служба Удаленный вызов процедур (RPC). Для работы данной службы нужна библиотека `WMIsvc.dll`.

Клиент отслеживания изменившихся связей

Служба предназначена для автоматического отслеживания связей (например, связей «ярлык—приложение») при перемещении объектов в пределах файловой системы компьютера или домена. Эта служба, например, реализует возможность автоматической корректировки пути, по которому ведет ярлык, если программа, на которую ссылается этот ярлык, будет перемещена. Если данная возможность вам не нужна, то можно отключить эту службу.

Служба Клиент отслеживания изменившихся связей занимает около 140 Кбайт оперативной памяти и запускается с правами локальной системы (Local System) автоматически при каждом входе пользователя в систему (запускается как часть процесса `svchost.exe`). Для отключения этой службы необходимо воспользоваться параметром `Start` из ветви реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\TrkWks`.

Для запуска службы Клиент отслеживания изменившихся связей необходимо, чтобы была запущена служба Удаленный вызов процедур (RPC). Для ее работы нужна библиотека `trkwks.dll`.

Модуль поддержки NetBIOS через TCP/IP

Служба реализует возможность трансляции NetBios-имени компьютера в его IP-адрес (для этого применяется WINS-сервер) и выполняет поддержку протокола NetBios поверх протокола TCP/IP. После отключения данной службы возможна некорректная работа функции доступа к сетевому компьютеру с помощью его NetBios-имени, поэтому не рекомендуется отключать эту службу, если компьютер подключен к сети.

Служба Модуль поддержки NetBIOS через TCP/IP занимает около 2700 Кбайт оперативной памяти и запускается с правами локальной службы (NT AUTHORITY\LocalService) автоматически при каждом входе пользователя в систему (запускается как часть процесса `svchost.exe`). Чтобы отключить эту службу, необходимо воспользоваться параметром `Start` из ветви реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\LmHosts`.

Для запуска службы Модуль поддержки NetBIOS через TCP/IP необходимо, чтобы были запущены службы AFD и NetBios через TCP/IP. Для ее работы нужна библиотека `lmhsvc.dll`.

Обозреватель компьютеров

Служба создает и обновляет список компьютеров, подключенных к сети. Данный список может быть необходим некоторым программам для своей работы, поэтому он автоматически обновляется службой через определенные интервалы времени. Если служба будет отключена, то список подключенных компьютеров создаваться не будет. Тем не менее это не повлияет на стабильность операционной системы, к тому же может повысить скорость открытия папки Сетевое окружение и общую скорость работы компьютера (если в сети присутствует большое количество компьютеров).

Служба Обозреватель компьютеров занимает около 70 Кбайт оперативной памяти и запускается с правами локальной системы (Local System) автоматически при каждом входе пользователя в систему (запускается как часть процесса `svchost.exe`). Чтобы отключить эту службу, необходимо воспользоваться параметром `Start` из ветви реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Browser`.

Для запуска службы Обозреватель компьютеров необходимо, чтобы были запущены следующие службы: Сервер и Рабочая станция. Для ее работы нужна библиотека `browser.dll`.

Оповещатель

Служба реализует возможность передачи сообщений другим сетевым компьютерам (пользователям, работающим на них), а также сообщений системы администратору компьютера. Если она будет остановлена, то программы, использующие ее

для передачи сообщений, работать не будут. Не будет также работать команда net send. Если данная команда вам не нужна, то можно отключить эту службу.

Служба Оповещатель занимает около 150 Кбайт оперативной памяти и запускается с правами локальной службы (NT AUTHORITY\LocalService) автоматически при каждом входе пользователя в систему (запускается как часть процесса svchost.exe). Чтобы отключить эту службу, необходимо воспользоваться параметром Start из ветви системного реестра HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Alerter.

Для запуска службы Обозреватель компьютеров необходимо, чтобы была запущена служба Рабочая станция. Для ее работы нужна библиотека alrsvc.dll.

Определение оборудования оболочки

Служба отвечает за автоматический запуск содержимого компакт-дисков или подключенной флэш-карты. После отключения этой службы из диалога свойств съемных дисков или компакт-дисков исчезнет вкладка Автозапуск, а также будет отключена сама возможность, реализуемая данной вкладкой. Отключение службы не влияет на возможность автозапуска дисков, за которую отвечает DWORD-параметр AutoRun из ветви реестра HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Cdrom. Если вы не используете возможность, предоставляемую вкладкой Автозапуск, то можете смело отключать эту службу.

Служба Определение оборудования оболочки занимает около 70 Кбайт оперативной памяти и запускается с правами локальной системы (Local System) автоматически при каждом входе пользователя в систему (запускается как часть процесса svchost.exe). Чтобы отключить эту службу, необходимо воспользоваться параметром Start из ветви системного реестра HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\ShellHWDetection.

Для запуска службы необходимо, чтобы была запущена служба Удаленный вызов процедур (RPC). Для ее работы нужна библиотека shsvcs.dll.

Планировщик заданий

С помощью данной службы реализуется возможность задания расписания, по которому будут регулярно запускаться те или иные программы. Неправильная настройка параметров данной службы может привести к появлению бреши в защите компьютера, поэтому настоятельно рекомендуется отключить эту службу.

Служба Планировщик заданий занимает около 250 Кбайт оперативной памяти и запускается с правами локальной системы (Local System) автоматически при каждом входе пользователя в систему (запускается как часть процесса svchost.exe). Чтобы отключить эту службу, необходимо воспользоваться параметром Start из ветви реестра HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Schedule.

Для запуска данной службы необходимо, чтобы была запущена служба Удаленный вызов процедур (RPC). Для ее работы нужна библиотека schedsvc.dll.

Рабочая станция

С помощью данной службы реализуется подключение к сетевым компьютерам, поэтому при ее остановке получить доступ к другим компьютерам не удастся. При отключении данной службы будут автоматически отключены службы Оповещатель, Сетевой вход в системы, Обозреватель компьютеров и др. Именно поэтому нельзя отключать эту службу при подключении компьютера к сети.

Служба Рабочая станция занимает около 70 Кбайт оперативной памяти и запускается с правами локальной системы (Local System) автоматически при каждом входе пользователя в систему (запускается как часть процесса `svchost.exe`). Чтобы отключить эту службу, необходимо воспользоваться параметром `Start` из ветви реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\lanmanworkstation`.

Для работы службы необходима библиотека `wkssvc.dll`.

Сервер

С помощью данной службы обеспечивается общий доступ к файлам, принтерам и именованным каналам данного компьютера. Иными словами, если данная служба будет отключена, то на компьютере нельзя будет создавать общедоступные ресурсы, а сетевые компьютеры не смогут получить к нему доступ. При отключении данной службы вам не удастся изменить или просмотреть группы, к которым принадлежат пользователи компьютера, с помощью оснастки `lusrmgr.msc`. Если данный компьютер не подключен к сети или не должен предоставлять общий доступ, то данную службу можно отключить.

Служба Сервер занимает около 120 Кбайт оперативной памяти и запускается с правами локальной системы (Local System) автоматически при каждом входе пользователя в систему (запускается как часть процесса `svchost.exe`). Чтобы отключить эту службу, необходимо воспользоваться параметром `Start` из ветви реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\lanmanserver`.

Для работы службы необходима библиотека `srvsvc.dll`.

Сетевой вход в систему

Служба поддерживает возможность входа в систему данного компьютера с помощью сетевого компьютера, входящего в домен, как будто пользователь входит локально. Если данный компьютер не входит в домен, не подключен к сети или сетевой вход в оболочку компьютера нежелателен, то лучше отключить эту службу.

Служба Сетевой вход в систему занимает около 2000 Кбайт оперативной памяти и запускается с правами локальной системы (Local System) автоматически при каждом входе пользователя в систему (запускается как часть процесса `svchost.exe`). Чтобы отключить эту службу, необходимо воспользоваться параметром `Start` из ветви реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Netlogon`.

Для запуска службы необходимо, чтобы была запущена служба Рабочая станция.

Служба восстановления системы

Служба является частью программы Восстановление системы (файл rstrui.exe, расположенный в каталоге %systemroot%\SYSTEM32\Restore). С помощью этой программы можно произвести откат к предыдущим настройкам операционной системы, если после установки какой-нибудь программы или драйвера система стала функционировать неверно или перестала загружаться в обычном режиме. Можно также самостоятельно создавать точки отката, чтобы потом была возможность вернуть настройки реестра и важных системных файлов Windows в состояние, которое они имели во время создания точки отката. Одним из минусов, который отпугивает пользователей от этой программы, является свободное пространство жесткого диска в размере 200 Мбайт, которое данная программа будет отбирать для своих нужд. И тем не менее не стоит сразу отключать данную программу — лучше некоторое время попользоваться ею, а уже потом осознанно решить, нужна она вам или нет.

Служба восстановления системы запускается с правами локальной системы (Local System) автоматически при каждом входе пользователя в систему (запускается как часть процесса svchost.exe). Чтобы отключить эту службу, необходимо воспользоваться параметром Start из ветви реестра HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\srservice.

Для запуска службы необходимо, чтобы была запущена служба Удаленный вызов процедур (RPC). Для ее работы нужна библиотека srsvc.dll.

Служба времени Windows

Служба реализует функцию синхронизации системного времени локального компьютера с сервером времени. На домашних компьютерах эту функцию используют не часто, поэтому ее можно смело отключать.

Служба времени Windows занимает около 100 Кбайт оперативной памяти и запускается с правами локальной системы (Local System) автоматически при каждом входе пользователя в систему (запускается как часть процесса svchost.exe). Для отключения этой службы необходимо воспользоваться параметром Start из ветви реестра HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\W32Time.

Для работы этой службы необходима библиотека w32time.dll.

Служба индексирования

Служба предназначена для индексации содержимого файлов на локальном диске с целью быстрого поиска при помощи оснастки ciadv.msc. В следующей главе будет подробно рассмотрена данная оснастка и работа с ней, а пока поговорим о службе. Служба индексирования запускается вручную (это единственная запускающаяся вручную служба, которая будет рассмотрена) с правами локальной системы (как процесс cisvc.exe). При этом если вы никогда не пользовались возможностью поиска в содержимом индексированных файлов, то лучше отключить данную службу (для этого необходимо воспользоваться DWORD-параметром Start из ветви реестра HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\CiSvc). Иначе поговорим о параметрах реестра, которые влияют на работу данной

службы, ведь их значения могут повлиять на производительность компьютера, использующего данную службу.

Все настройки службы индексирования хранятся в ветви системного реестра HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\ContentIndex. Большая часть находящихся здесь параметров имеет тип DWORD, поэтому будем считать, что если о типе параметра ничего не сказано, значит, он — DWORD.

- DaemonResponseTimeout — определяет интервал времени перед повторной попыткой индексирования содержимого поврежденного файла. Можно попробовать установить значение равным 2.
- DelayedFilterRetries — указывает количество попыток переиндексации содержимого файла, если предыдущая попытка его индексации завершилась неудачно. Если вам не так важно, чтобы служба индексирования включала в себя сведения о содержимом всех файлов, лучше этому параметру присвоить значение от 1 до 3, снизив тем самым возможное время индексации и нагрузку на процессор.
- DelayUsnReadOnLowResource — если значение равно 1, то при нехватке системных ресурсов поле Update Sequence Number читаться не будет.
- EventLogFlags — если значение равно 0, то в журнале событий, происходящих при индексировании содержимого, регистрации не будет. Если значение равно 2, то будут регистрироваться лишь ошибки фильтрации внедренных объектов. Рекомендуется некоторое время понаблюдать за ошибками индексации и, если частых ошибок в работе индексации не будет, лучше запретить регистрацию ошибок индексации.
- FilterBufferSize — определяет размер в килобайтах буфера, используемого фильтром индексации. Не стоит забывать, что слишком большое значение этого параметра может повлиять на общую скорость работы компьютера в моменты индексации (по умолчанию индексация происходит довольно часто), ведь буфер является частью оперативной памяти компьютера. По умолчанию этот параметр не существует.
- FilterDelayInterval — указывает интервал времени в секундах, который служба индексации будет ожидать перед продолжением индексации, если в буфер фильтра индексации было помещено определенное количество документов (это время необходимо для очистки буфера).
- FilterDirectories — если значение равно 0, то свойства системных каталогов не индексируются. Любое другое значение говорит об индексации свойств.
- FilterFilesWithUnknownExstensions — если значение равно 0, то содержимое файлов с незарегистрированными расширениями индексироваться не будет. Любое другое значение говорит об индексировании этих файлов.
- FilterIdleTimeout — определяет интервал времени в миллисекундах, по истечении которого неиспользуемые библиотеки фильтра будут выгружаться.
- FilterRemainingThreshold — указывает количество документов в буфере, после преодоления которого начинается переиндексация содержимого локаль-

ных дисков. По умолчанию значение определяет 35 документов, но можно его увеличить.

- `FilterRetries` — определяет количество попыток повторного индексирования файлов. По умолчанию значение равно `4`, но если полнота индексированных файлов не так важна, то можно уменьшить его (параметр может принимать значения в пределах от `0` до `10`).
- `FilterRetryInterval` — указывает интервал ожидания в минутах, после которого система предпримет повторную попытку индексации содержимого документов, которые при предыдущей попытке использовались другими программами.
- `ForcedNetPathScanInterval` — определяет интервал времени, по истечении которого служба индексации начнет переиндексацию содержимого каталогов, даже если в нем не было изменений.
- `IsEnumAllowed` — если значение равно `0`, то перечисления при индексации использоваться не будут, что снижает нагрузку на сервер.
- `IsIndexingIMAPSvc` — определяет, будет ли выполняться индексирование содержимого почтовых сообщений IMAP. Если значение равно `0`, то данные сообщения индексироваться не будут.
- `IsIndexingNNTPSvc` — указывает, будет ли выполняться индексирование содержимого почтовых сообщений NNTP. Если значение равно `0`, то данные сообщения индексироваться не будут.
- `IsIndexingW3SVC` — определяет, будет ли индексироваться содержимое файлов сервера LLS. Если значение равно `0`, то индексирования не будет.
- `LowResourceSleep` — указывает интервал ожидания в секундах перед продолжением индексации, если ресурсов компьютера для продолжения индексации не хватает. Параметр может принимать значения в диапазоне от `5` до `1200`.
- `MaxFilesizeFiltered` — определяет максимальный размер файлов, которые допущены к индексации. Если размер файла превышает максимальный размер файла для индексации, то содержимое файла индексироваться не будет. По умолчанию значение равно 256 Кбайт.

Для запуска службы необходимо, чтобы была запущена служба Удаленный вызов процедур (RPC). Для корректной работы службы нужны все библиотеки, определенные в `REG_MULTI_SZ`-параметре `DLLsToRegister` из ветви системного реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\ContentIndex`.

Служба сообщений

Служба управляет возможностью передачи сообщений сетевым компьютерам. Если эта служба будет отключена, то программы, ее использующие, не смогут передавать сообщения. В частности, будет отключена команда `net send «êîìïüþòåð» «ñîîáùåíèå»`, используемая системой для передачи администратору сообщений счетчиков производительности и т. д. Если команда `net send` вам не нужна, то данную службу можно отключить.

Служба сообщений занимает около 70 Кбайт оперативной памяти и запускается с правами локальной системы (Local System) автоматически при каждом входе

пользователя в систему (запускается как часть процесса `svchost.exe`). Чтобы отключить эту службу, необходимо воспользоваться параметром `Start` из ветви реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Messenger`.

Для запуска службы необходимо, чтобы были запущены службы Удаленный вызов процедур (RPC), Рабочая станция, Интерфейс NetBIOS и Plug and Play. Для работы службы нужна библиотека `msgsvc.dll`.

Службы IPSEC

Служба управляет политикой безопасности IP-протокола, а также запускает ISAKMP/Oakley (IKE) и драйвер IP-безопасности. Благодаря этой службе возможно использование протокола IPSec для защиты и шифрования данных, передаваемых протоколом IP. При этом для реализации защищенной передачи данных по протоколу IP необходимо, чтобы служба была запущена на обоих компьютерах, участвующих в соединении (а также соответствующим образом настроена). Как правило, в домашних сетях нет необходимости в этой службе.

Служба занимает около 900 Кбайт оперативной памяти и запускается с правами локальной системы (Local System) автоматически при каждом входе пользователя в систему (запускается как часть процесса `lsass.exe`). Чтобы отключить эту службу, необходимо воспользоваться параметром `Start` из ветви реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\PolicyAgent`.

Для запуска данной службы необходимо, чтобы были запущены следующие службы: Удаленный вызов процедур (RPC), Драйвер IPSEC и Драйвер протокола TCP/IP.

Службы терминалов

Служба предоставляет возможность работы с программами терминалов, а также позволяет нескольким пользователям одновременно интерактивно подключаться к компьютеру и отображать удаленный Рабочий стол. Как правило, если компьютер не подключен к сети или к нему не должны получать доступ из сети, эту службу лучше отключить.

Служба терминалов занимает около 900 Кбайт оперативной памяти и запускается с правами локальной системы (Local System) автоматически при каждом входе пользователя в систему (запускается как часть процесса `svchost.exe`). Для отключения этой службы необходимо воспользоваться параметром `Start` из ветви реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\TermService`.

Для запуска службы необходимо, чтобы была запущена служба Удаленный вызов процедур (RPC). Для работы службы нужна библиотека `termsrv.dll`.

Справка и поддержка

С помощью данной службы реализована возможность работы Центра справки и поддержки. Если служба будет отключена, то запуск Центра справки и поддержки будет невозможен. Если же этот центр вам не нужен, то можно отключить службу.

Служба Справка и поддержка занимает около 100 Кбайт оперативной памяти и запускается с правами локальной системы (Local System) автоматически при каждом входе пользователя в систему (запускается как часть процесса `svchost.exe`). Чтобы отключить эту службу, необходимо воспользоваться параметром `Start` из ветви реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\helpsvc`.

Для запуска службы необходимо, чтобы была запущена служба Удаленный вызов процедур (RPC). Для работы службы нужна библиотека `pchsvc.dll`.

Удаленный реестр

Служба управляет возможностью доступа к реестру данного компьютера с помощью другого компьютера. Если данная служба будет отключена, то получить доступ к реестру данного компьютера можно будет только локально. Как правило, эту службу необходимо отключить, так как неправильная настройка может привести к снижению уровня безопасности компьютера.

Служба Удаленный реестр занимает около 60 Кбайт оперативной памяти и запускается с правами локальной службы (NT AUTHORITY\LocalService) автоматически при каждом входе пользователя в систему (запускается как часть процесса `svchost.exe`). Чтобы отключить эту службу, необходимо воспользоваться параметром `Start` из ветви системного реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\RemoteRegistry`.

Для запуска службы необходимо, чтобы была запущена служба Удаленный вызов процедур (RPC). Для работы службы нужна библиотека `regsvc.dll`.

Файловая система

Настройки файловой системы являются критически важными для оптимизации скорости работы компьютера. Как правило, если параметры реестра, описывающие работу файловой системы, не оптимизированы, то возможно снижение скорости работы компьютера на 10–70 %. Все настройки файловой системы Windows XP расположены в ветви реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\FileSystem` (в основном для хранения настроек используются параметры `DWORD`-типа). Рассмотрим некоторые из этих параметров.

ВНИМАНИЕ

Как уже говорилось, настройки файловой системы являются критически важными, поэтому перед их изменением настоятельно рекомендуется сделать копию файлов реестра. Другой рекомендацией является поочередное изменение параметров файловой системы. Лучше всего изменить сначала значение одного параметра, после этого некоторое время поработать в системе (обязательно необходимо перезагрузить компьютер, чтобы изменения вступили в силу). Если после изменения параметра система работает стабильно и скорость работы при этом не уменьшилась, то можно попробовать изменить значение другого параметра.

- `ConfigFileAllocSize` — определяет минимальный размер файла (в килобайтах), для которого при его создании выполняется поиск оптимального места на диске. Под оптимальным понимается то место файловой системы, для записи в которое нет необходимости во фрагментации записываемого файла. По умолчанию данный параметр отсутствует в реестре.
- `Win95TruncatedExtensions` — если значение равно `1`, то при сравнении расширений файлов, расширения, состоящие больше чем из трех символов, будут усекаться до трех. Это может быть необходимо для совместимости с предыдущими версиями Windows.
- `NtfsDisable8dot3NameCreation` — если значение равно `1`, то система не будет генерировать имена файлов в формате 8.3 (восемь символов на имя файла и три символа на расширение файла), используемом в операционной системе MS-DOS.
- `NtfsDisableLastAccessUpdate` — определяет, будет ли происходить обновление метки последнего доступа к файлам. Использование метки может понизить скорость открытия папок, и, как правило, оно не нужно пользователям, поэтому метку последнего доступа можно отключить. Если значение равно `1`, то запись метки последнего доступа будет отключена. По умолчанию не существует.
- `NtfsEncryptionService` — этот параметр строкового типа указывает имя службы, которая обеспечивает шифрование файлов в системе NTFS. По умолчанию не существует.
- `NtfsMftZoneReservation` — определяет количество места, занимаемого на жестком диске главной файловой таблицей. Параметр указывает не размер занимаемого места, а условие, на основе которого система сама выберет необходимый размер. При этом параметр может принимать следующие значения:
  - `1` — файловая система будет настроена для хранения малого количества файлов, которые при этом будут иметь большой размер;
  - `4` — файловая система будет настроена для хранения большого количества файлов, которые при этом будут иметь маленький размер;
  - значения в пределах от `1` до `4` являются промежуточными.

  По умолчанию параметр не существует.

Приоритеты прерываний и процессов первого плана

Существует возможность указания системе приоритета для конкретного прерывания. В зависимости от использования прерывания повышение его приоритета может повысить скорость работы компьютера. Можно также указать приоритет процесса первого плана (то есть программы, с которой пользователь работает в данный момент). Обе эти возможности реализуются с помощью ветви реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\PriorityControl`, содержащей следующие параметры `DWORD`-типа.

- `IRQ08Priority` — если значение равно `1`, то восьмое прерывание будет считаться приоритетным (как правило, восьмое прерывание является прерыванием

CMOS и часов), что может повысить скорость работы компьютера. Можно также изменить и приоритетность других прерываний. Для этого достаточно создать этот параметр с указанием другого прерывания; например для прерывания 9 (ACPI-совместимый компьютер) необходимо создать параметр `IRQ09Priority` и присвоить ему значение `1`.

ПРИМЕЧАНИЕ

Номера используемых прерываний IRQ и устройств, их использующих, можно посмотреть в Диспетчере устройств. Для этого достаточно открыть диалог свойств необходимого устройства и перейти на вкладку Ресурсы.

- `Win32PrioritySeparation` — определяет длительность кванта времени для программы переднего плана. Благодаря этому повышается производительность работы программы, с которой в данный момент работает пользователь. Данный параметр может принимать следующие значения:
  - `0` — каждый квант времени для программы, работающей на переднем плане, увеличивается в 6 раз;
  - `1` — квант времени увеличивается в 12 раз;
  - `2` — квант увеличивается в 18 раз. Данное значение параметра используется по умолчанию.

ПРИМЕЧАНИЕ

Квант — это единица процессорного времени, выделяемая программе для работы. По истечении данного времени произойдет переключение на работу другого приложения, если на компьютере запущено несколько программ. При этом программа, использовавшая свой квант времени, перемещается в конец очереди на повторное его получение. Благодаря этому возможна реализация многозадачности, то есть программы работают по очереди, в течение своего кванта времени, но кажется, что они работают одновременно.

Количество оперативной памяти, используемой файловой системой

При работе компьютера файловая система Windows XP резервирует определенный размер оперативной памяти для операций I/O (операций чтения/записи). Чем больше будет размер резервируемой памяти, тем быстрее будут происходить такие операции файловой системы, как открытие папок и файлов, перемещение, копирование и удаление файлов и папок.

По умолчанию используются следующие размеры резервируемой памяти:

- для 32 Мбайт оперативной памяти — 4 Кбайт;
- 64 Мбайт оперативной памяти — 8 Кбайт;

- 128 Мбайт оперативной памяти — 16 Кбайт;
- 256 Мбайт оперативной памяти — 64 Кбайт;
- 512 Мбайт оперативной памяти — 128 Кбайт.

Можно также самостоятельно указать количество используемой для операций I/O памяти. Для этого применяется DWORD-параметр `IOPageLockLimit`, расположенный в ветви реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Session Manager\Memory Management`. По умолчанию не существует.

Для повышения производительности системы можно воспользоваться DWORD-параметром `LargeSystemCache`, расположенным в ветви системного реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Session Manager\Memory Management`. Если значение этого параметра равно 1, то 4 Мбайт оперативной памяти будет дополнительно зарезервировано для кэша файловой системы.

Альтернативные подсистемы

Кроме подсистемы Windows, операционная система Windows XP поддерживает и другие подсистемы. Например, к поддерживаемым ею подсистемам относится Posix. Она очень редко применяется пользователями и тем не менее постоянно отбирает некоторый объем оперативной памяти. Если вы точно знаете, что никогда не будете пользоваться этой подсистемой, то можно удалить упоминания о ней в реестре (или просто изменить названия параметров, в которых это упоминание записано), после чего занимаемая подсистемой оперативная память будет освобождена.

Список поддерживаемых операционной системой Windows XP подсистем расположен в ветви реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Session Manager\SubSystems`. Ветвь может хранить следующие параметры строкового типа.

- `Optional` — определяет подсистему Posix и содержит значение `Posix`. Если вы не используете эту систему, то можно изменить название параметра, чтобы он больше не применялся.
- `Posix` — указывает файл, используемый для реализации работы подсистемы Posix. Если вы не используете эту систему, то можно изменить название параметра, чтобы он больше не применялся.

Диски

Отдельно хотелось бы сказать о некоторых настройках дисков, а также о настройках обслуживания дисков. Например, об автоматической дефрагментации дисков или о настройке скорости работы жестких дисков. В основном речь в этом разделе пойдет о жестких дисках, хотя несколько слов будет сказано и о гибких.

Автоматическая дефрагментация

Одним из нововведений Windows XP является автоматическая дефрагментация файловой системы при простое компьютера в течение определенного промежутка време-

ни (10–30 минут). При этом по умолчанию также выполняется дефрагментация загрузочного диска и файлов, необходимых для загрузки компьютера. Настройки автоматической дефрагментации расположены в ветви реестра HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Dfrg\BootOptimizeFunction. Ветвь может содержать следующие параметры строкового типа.

- Enable — определяет, включена ли функция автоматической дефрагментации файловой системы диска. По умолчанию его значение равно Y, то есть дефрагментация включена. Если присвоить этому параметру значение N, то дефрагментация будет запрещена.
- OptimizeComplete — показывает, была ли успешной предыдущая попытка дефрагментации. Если значение равно yes, то дефрагментация была успешной. Если же значение равно no, то дефрагментация выполнена не была. При этом причина, по которой не была выполнена дефрагментация, будет записана в следующий рассматриваемый параметр.
- OptimizeError — определяет причину, по которой не была выполнена дефрагментация при предыдущей попытке ее выполнения.

Проверка диска при неправильном выключении компьютера

По умолчанию при неправильном выключении компьютера перед запуском операционной системы проверяется состояние файловой системы жестких дисков. Настройки этой проверки также можно изменить. Например, можно изменить время, которое ожидает операционная система перед тем, как начать проверку. По умолчанию операционная система в течение 10 секунд ожидает решения пользователя — начать проверку или отменить ее. Количество секунд ожидания системы определено в DWORD-параметре AutoChkTimeout, расположенном в ветви реестра HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Session Manager. Если значение данного параметра равно 0, то система будет начинать проверку жесткого диска немедленно, не ожидая решения пользователя. Остальные значения определяют количество секунд, которые система будет ожидать ответа. По умолчанию параметр не существует.

В этой же ветви реестра расположен параметр REG_MULTI_SZ-типа, имеющий название BootExecute. Его значение определяет список системных служб, которые будут запускаться при включении компьютера. По умолчанию параметр содержит лишь строку autocheck autochk *, которая как раз и запускает службу проверки жестких дисков. Другими словами, если удалить эту строку, то при неверном выключении компьютера проверка жестких дисков выполняться не будет.

Режимы работы жестких дисков

Одной из проблем, которая может произойти с жесткими дисками UltraATA, является неверное определение скоростного режима, в котором может работать жесткий диск. Вследствие этого жесткий диск будет работать на заведомо меньшей скорости, чем максимально поддерживаемая им. Чтобы решить эту проблему, необходимо воспользоваться реестром — нужные сведения расположены в ветви реестра HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\

Class\{4D36E96A-E325-11CE-BFC1-08002BE10318}. Ветвь содержит подразделы, имеющие следующий формат — 0000, 0001, 0002. Каждый из них определяет настройки одного контроллера жесткого диска. При этом если в системе находится только один жесткий диск, то, как правило, сведения о нем хранятся в подразделе 0000. Этот подраздел может включать в себя следующие параметры DWORD-типа.

- EnableUDMA66 — определяет, включен ли режим UDMA66 для данного диска. Если ваш диск поддерживает этот режим, но при этом работает в режиме UDMA33, то стоит попробовать воспользоваться этим параметром для перехода на более быстрый режим.
- MasterDeviceTimingModeAllowed и SlaveDeviceTimingModeAllowed — также определяют режим работы жестких дисков. Первый из них отвечает за канал Master, а второй — за канал Slave. Параметры указывают, в каком максимальном режиме могут работать диски. Если значения этих параметров равны 0õffffffff, то диск может работать в режиме UDMA, а если значения равны 0õ0000001f, то только в PIA.
- MasterDeviceTimingMode и SlaveDeviceTimingMode — если ваш жесткий диск может работать в режиме UDMA, то эти параметры определят сам режим, который будет использоваться. Вот некоторые значения, которые могут принимать данные параметры:
  - 0x00010010 — жесткий диск использует режим UDMA Mode 5 (ATA100);
  - 0x000fffff — режим UDMA Mode 5 (ATA100);
  - 0x00008010 — режим UDMA Mode 4 (ATA66);
  - 0x0000ffff — режим UDMA Mode 4 (ATA66);
  - 0x00002010 — режим UDMA Mode 2 (ATA33);
  - 0x00000410 — режим Multi-Word DMA Mode 2 и PIO 4.

Сообщения о недостатке свободного места на диске

По умолчанию если на диске остается меньше 10 % свободного места, то система выдает сообщение об этом и просит очистить диск от ненужных программ. В наше время, когда жесткие диски объемом в 200 Гбайт не редкость, порог в 10 % уже неактуален — смешно получать сообщение с просьбой очистки диска, если на нем еще остается 20 Гбайт свободного места. С помощью реестра существует возможность либо вообще заблокировать отображение таких сообщений, либо уменьшить порог свободного места, при котором эти сообщения начнут выдаваться.

Чтобы определить порог выдачи сообщений о нехватке места, необходимо воспользоваться DWORD-параметром DiskSpaceThreshold, расположенным в ветви реестра HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\lanmanserver\parameters. Значение этого параметра определяет процент занимаемого на диске места, преодоление которого станет стартовым моментом в выдаче сообщений о нехватке места. Следовательно, в шестнадцатеричной сис-

теме данный параметр может принимать значения в диапазоне от 0 до 64. По умолчанию не существует.

Можно вообще запретить выдачу сообщений о нехватке места. Для этого применяется DWORD-параметр NoLowDiskSpaceChecks, расположенный в ветви реестра HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\policies\Explorer. Если значение этого параметра равно 1, то выдача сообщений о нехватке места на диске будет прекращена. По умолчанию не существует.

Другие настройки

К другим настройкам операционной системы можно отнести настройки автоматического завершения работы программ и служб при завершении работы компьютера, выгрузки библиотек при завершении работы программ, а также некоторые другие настройки, которые будут рассмотрены.

Автоматическое завершение программ при выключении компьютера

Перед завершением работы компьютера операционная система дает всем программам и службам, запущенным в данный момент, приказ на сохранение своих данных и завершение работы. После этого операционная система ожидает некоторое время, пока не будут завершены все программы и службы. Если в течение этого времени какая-то программа или служба не завершила свою работу, эта программа или служба считается зависшей. По умолчанию операционная система не пытается автоматически завершить работу зависшего процесса. Вместо этого она выводит для пользователя диалог, в котором говорится, что определенная программа или служба не отвечает, и пользователю задается вопрос, что же необходимо делать дальше — либо в принудительном порядке завершить работу зависшей службы или программы, либо подождать некоторое время, пока она сама завершит свою работу.

Если данный диалог выводится очень редко, то с ним еще можно смириться. Но если он выводится практически при каждом завершении работы системы? Тогда реестр Windows позволяет задать для завершения работы системы режим автоматического завершения работы зависших программ. В этом случае, если какая-то программа или служба не отвечает в течение 20 секунд, которые будут даваться на завершение работы процессов, то работа зависшей программы будет завершаться автоматически. При этом следует учитывать, что, поскольку выполняется принудительное завершение работы процесса, существует вероятность, что все данные, которыми в это время оперировал зависший процесс, будут утеряны.

Чтобы заставить операционную систему при выходе автоматически завершать работу процессов, которые не завершили свою работу в течение определенного времени (по умолчанию 20 секунд), необходимо присвоить параметру строкового типа AutoEndTasks значение 1. Он расположен в ветви реестра HKEY_CURRENT_USER\Control Panel\Desktop.

Если значение параметра AutoEndTasks равно 1, то можно определить значение еще нескольких параметров, которые влияют на процесс автоматического завершения работы программ и служб. К ним можно отнести следующие параметры строкового типа.

- HungAppTimeout — определяет интервал времени (в миллисекундах), в течение которого программа должна ответить на запрос системы. Если в течение этого времени программа не ответила на запрос, то она считается зависшей. Значение не используется при завершении работы компьютера — параметр предназначен для определения зависших программ в процессе работы пользователя в системе. При этом система не предпринимает никаких действий по завершению работы зависшей программы, даже если значение параметра AutoEndTasks равно 1.

  Параметр расположен в ветви реестра HKEY_CURRENT_USER\Control Panel\Desktop. По умолчанию его значение равно 5000, что соответствует интервалу ожидания ответа, равному 5 секунд. В принципе, значение данного параметра можно уменьшить до 2000, что соответствует 2 секундам. Хотя никакой функциональной разницы от этого изменения не будет — просто на три секунды раньше программа станет считаться зависшей. При этом если программа все-таки ответит на запрос системы, то она опять станет считаться работающей.

- WaitToKillAppTimeout — указывает интервал времени (в миллисекундах), в течение которого программа должна закончить свою работу при завершении работы операционной системы. Если программа не завершит свою работу в течение указанного интервала времени, то она будет завершена автоматически. При этом все несохраненные данные будут утеряны.

  Параметр расположен в ветви реестра HKEY_CURRENT_USER\Control Panel\Desktop. По умолчанию его значение равно 20000, что соответствует интервалу ожидания ответа, равному 20 секунд. Этого интервала с избытком хватает на завершение работы программ при завершении работы операционной системы. Более того, если во время завершения работы операционной системы довольно редко зависают программы, то значение параметра WaitToKillAppTimeout можно уменьшить до 5000 — интервала в 5 секунд, как правило, также всегда хватает для корректного завершения работы программ.

- WaitToKillServiceTimeout — расположен в ветви системного реестра HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control. Его значение определяет интервал времени (в миллисекундах) в течение которого служба должна закончить свою работу при завершении работы операционной системы. По умолчанию значение этого параметра равно 20000. Этого также с избытком хватает для завершения работы служб, но можно присвоить этому параметру значение, равное, например, 7000. Ниже этого значения лучше не опускаться, ведь, как часто бывает, службы оперируют информацией, критичной для корректной работы операционной системы (тем более, большая часть служб запущена с правами системы, то есть может изменять значения многих ветвей реестра), некорректная запись которой может привести к повреждению реестра.

Выгрузка библиотек при выходе из программы

По умолчанию система при завершении работы программы оставляет в памяти ее библиотеки на случай, если через некоторое время она снова будет запущена. С одной стороны, плюс этого метода очевиден — следующий запуск будет выполнен быстрее. Но, с другой стороны, очевиден и минус этого метода — библиотеки программы занимают место в оперативной памяти, которого иногда и так катастрофически не хватает.

Настройками выгрузки библиотек программ руководит параметр DWORD-типа AlwaysUnloadDll, расположенный в ветви реестра HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer. По умолчанию значение этого параметра равно 0 — это говорит о том, что библиотеки не будут выгружаться вместе с программой. Если же необходимо выгружать библиотеки, то следует присвоить этому параметру значение 1.

Функция упреждающей выборки

Одним из нововведений операционной системы Windows XP стала возможность упреждающей выборки (prefetching) часто используемых программ. Иными словами, при первом запуске программы (и при запуске операционной системы) части ее кода копируются в специальную папку (%systemroot%\prefetch), а при следующих запусках программы эти части сразу берутся из папки, тем самым ускоряя запуск программы. Эта возможность включена по умолчанию.

Настройки упреждающей выборки расположены в ветви системного реестра Windows HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Session Manager\Memory Management\PrefetchParameters. При этом параметром, управляющим работой функции упреждающей выборки, является DWORD-параметр EnablePrefetcher. Он может содержать следующие значения.

- 0 — полностью отключить механизм упреждающей выборки. Как правило, данный механизм действительно повышает загрузку программ, поэтому отключать его не стоит.
- 1 — задействовать только возможность упреждающей выборки запускаемых приложений. При этом возможность упреждающего чтения при запуске компьютера использоваться не будет — как правило, эта возможность понижает скорость загрузки компьютера, поэтому, если критерий времени загрузки компьютера важен, следует использовать именно это значение параметра.
- 2 — задействовать только возможность упреждающей выборки при запуске компьютера. Возможность упреждающей выборки запускаемых приложений использоваться не будет.
- 3 — задействовать обе возможности. Это значение присвоено по умолчанию.

Настройки поведения при крахе системы или программы

При крахе операционной системы или программы (под крахом понимается ошибка, после появления которой работа программы или системы аварийно останавливается) задействуется сразу несколько механизмов операционной системы Windows XP.

К ним можно отнести запись ошибки в журнал событий (`eventvwr.msc`), попытку выполнения отладки программы, отправку сведений об ошибке корпорации Microsoft, создание дампа памяти, если произошел крах операционной системы.

Подробнее работа журнала событий будет рассмотрена при описании консоли управления Microsoft, сейчас же будут описаны параметры реестра, относящиеся к отправке корпорации Microsoft сообщений о возникшей ошибке. Механизм отправки сообщений об ошибках задумывался как способ поиска и устранения ошибок в программах корпорации Microsoft. При этом попытка отправки отчета об ошибке происходит даже в том случае, когда компьютер не подключен к Интернету. Именно поэтому иногда необходимо отключить отправку сообщений об ошибках. Например, когда компьютер не имеет доступа к Интернету или когда нет желания платить за дополнительный трафик.

Все настройки отправки сообщений расположены в ветви системного реестра `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\PCHealth\ErrorReporting`. Она содержит как параметры, имеющие тип `DWORD`, так и подразделы. Среди параметров, которые присутствуют в этой ветви, можно выделить следующие.

- `AllOrNone` — с его помощью можно определить, сообщения об ошибках в каких программах будут отправляться Microsoft. Если значение этого параметра равно `1`, то будут отправляться сообщения об ошибках во всех приложениях, установленных на компьютере. Если же значение этого параметра равно `0`, то сообщения об ошибках будут отправляться только в приложениях, перечисленных в подразделе реестра `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\PCHealth\ErrorReporting\InclusionList`. При этом также можно определить приложения, сообщения об ошибках в которых не будут отправляться корпорации Microsoft. Для этого предназначена ветвь реестра `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\PCHealth\ErrorReporting\ExclusionList`. Оба приведенных подраздела реестра должны содержать параметры `DWORD`-типа с именами, соответствующими определенной программе (например, для описания программы Проигрыватель Windows Media имя параметра должно быть равно `wmplayer.exe`). Значения этих параметров должны быть равны `1`.
- `DoReport` — определяет, будет ли задействован механизм отправки сообщений о крахе программы Microsoft. Если значение этого параметра равно `0`, то при возникновении ошибки в приложении никаких отчетов корпорации Microsoft отправляться не будет.
- `DoTextLog` — при установке значения равным `0` не будут записываться сообщения о крахе программ в журнал. По умолчанию параметр не существует.
- `IncludeKernelFaults` — если значение равно `0`, то при крахе операционной системы не будет происходить попытка отправки сообщения Microsoft. Значение этого параметра используется, только если значение параметра `DoReport` равно `1`, а значение параметра `ShowUI` равно `3`.
- `IncludeMicrosoftApps` — если значение равно `0`, то при крахе программ, созданных Microsoft, не будет происходить попытка отправки сообщения Microsoft.

Значение данного параметра используется, только если значение параметра DoReport равно 1, а значение параметра ShowUI равно 3.

- IncludeWindowsApps — если значение равно 0, то при крахе компонентов операционной системы Windows XP не будет происходить попытка отправки сообщения Microsoft. Значение данного параметра используется, только если значение параметра DoReport равно 1, а значение параметра ShowUI равно 3.

Отдельно можно определить настройки отправки отчетов об ошибках в продуктах из комплекта Office. Для этого предназначена ветвь реестра HKEY_CURRENT_USER\Software\Microsoft\Office\10.0\Common. Она содержит следующие параметры DWORD-типа.

- DWNoExternalURL — если значение равно 1, то соединение с сервером Microsoft для отправки сообщений об ошибках будет запрещено.
- DWNoFileCollection — при установке значения равным 1 будет запрещена отправка файлов, запрашиваемых сервером обработки ошибок (иногда серверу обработки ошибок необходимы дополнительные файлы для определения причины возникшей ошибки, в этом случае он отправляет запрос на передачу этих файлов).
- DWNoSecondLevelCollection — если значение равно 1, то будет запрещена отправка серверу обработки ошибок файлов, используемых программой, в которой произошла ошибка, а также содержимого реестра компьютера пользователя.
- DWNeverUpload — при установке значения равным 1 будет запрещена загрузка файлов на удаленный компьютер.

По умолчанию параметры не существуют.

Кроме генерации отчетов об ошибках, операционная система по умолчанию запускает стандартный отладчик drwtsn.exe для попытки восстановления работы программы. Если эта возможность вам также не нужна (не многие пользователи в наше время знакомы с «Ассемблером»), то можно отключить запуск отладчика. Для этого достаточно параметру строкового типа Auto присвоить значение 0. Он расположен в ветви реестра HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\AeDebug. В данной ветви также присутствует параметр строкового типа Debugger, который определяет путь к программе для отладки приложения.

Если же произошел крах системы, после которого нормальная работа операционной системы невозможна, то система записывает в файл дамп памяти. В этом дампе находится содержимое памяти, вызвавшее неустранимую ошибку. После записи дампа памяти система перезагружает компьютер или выводит экран BSOD («синий экран смерти»). Как правило, обычным пользователям записываемый дамп памяти совершенно не нужен — что с ним делать? Поэтому можно отключить запись дампа памяти. Настройки поведения системы при аварийной остановке

находятся в ветви реестра HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\CrashControl. Она включает в себя следующие параметры.

- CrashDumpEnabled — этот параметр DWORD-типа определяет, будет ли система записывать дамп памяти, и если да, то какой размер дампа для этого будет использоваться. Если значение параметра равно 0, то запись дампа памяти при ошибке будет запрещена. Если значение равно 1, то при аварийной остановке будет создан малый дамп памяти (размером 64 Кбайт). Если значение равно 2, то при ошибке будет создаваться дамп памяти ядра. Если же значение равно 3, то будет создаваться файл, содержащий полный дамп памяти (равный объему установленной в системе оперативной памяти).
- DumpFile — параметр REG_EXPAND_SZ-типа указывает путь к файлу (и название файла), в который будет записываться полный дамп памяти (если значение параметра CrashDumpEnabled равно 3). Например, по умолчанию значение равно %systemRoot%\Memory.dmp.
- KernelDumpOnly — если значение данного параметра DWORD-типа равно 1, то при ошибке в работе операционной системы в журнал отладки будет записываться только информация о состоянии ядра операционной системы при ошибке. Если же значение равно 0, то в журнал будут заноситься не только сведения о состоянии ядра, но и сведения о состоянии памяти и всех остальных устройств, установленных на компьютере. По умолчанию параметр не существует.
- LogEvent — при установке значения этого параметра DWORD-типа равным 0 в системный журнал не будут записываться сведения о произошедших ошибках в работе операционной системы.
- MinidumpDir — этот параметр REG_EXPAND_SZ-типа определяет путь к каталогу, в который будут записываться файлы малого дампа, если значение параметра CrashDumpEnabled равно 1. Стоит заметить, что в случае использования создания полного дампа памяти будет происходить запись дампа в единственный файл — при каждой новой ошибке данный файл будет переписываться. Если же используются малые дампы памяти, то для каждой ошибки будет создаваться свой файл малого дампа памяти. По умолчанию значение параметра равно %SystemRoot%\Minidump.
- Overwrite — параметр DWORD-типа, указывает, будет ли переписываться файл журнала, если достигнут предел этого файла. Если значение равно 1, то при достижении предела файла журнала этот файл будет удален, а на его месте будет создан новый файл. Если же значение равно 0, то в файл журнала не будет записываться информация об ошибке, если он уже существует. По умолчанию значение этого параметра равно 1.
- SendAlert — этот параметр DWORD-типа определяет, будет ли послано сообщение о произошедшей ошибке администратору. По умолчанию значение равно 1, то есть сообщения администратору отсылаться будут.

По умолчанию, если возникает ошибка оболочки Windows XP, система автоматически перезагружает оболочки. За это отвечает DWORD-параметр AutoRestartShell, расположенный в ветви реестра HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon. Если его значение равно 1, то пере-

загрузка оболочки при ошибке в ней будет выполняться. Существует также параметр, определяющий, будет ли выполняться отладка процесса csrss.exe, если в нем произойдет ошибка. Он находится в той же ветви, что и ранее рассмотренный параметр, и называется DebugServerCommand (имеет строковый тип). По умолчанию значение этого параметра равно no, то есть данный процесс не отлаживается. Чтобы разрешить отладку этого процесса, необходимо присвоить параметру значение yes. Можно также определить, будет ли автоматически перезагружаться компьютер при аварийной остановке операционной системы («синий экран смерти», BSOD). Для этого используется DWORD-параметр AutoReboot, расположенный в ветви системного реестра HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\CrashControl. Если значение этого параметра равно 1, то компьютер будет автоматически перезагружаться. Если же значение равно 0, то при аварийной остановке системы будет выводиться «синий экран смерти».

Размер кэша второго уровня

Существует возможность указания размера кэша второго уровня процессора, который установлен в системе. Для этого применяется параметр DWORD-типа SecondLevelDataCache. Если его значение равно 0, то размер кэша будет определяться HAL автоматически (если это сделать не получится, то будет считаться, что кэш второго уровня имеет размер 256 Кбайт). Если же данный параметр имеет отличное от нуля значение, то оно и будет равно размеру кэша второго уровня в байтах.

Сеть и сетевые компоненты

Отдельно хотелось бы сказать о параметрах реестра, относящихся к настройке сетевых компонентов операционной системы Windows. Их очень много, поэтому для описания всех параметров не хватит одной главы — для этого нужна целая книга. Здесь же будут рассмотрены наиболее интересные параметры, с помощью которых можно настроить различные возможности работы протоколов и стеков протоколов, а также отдельных сетевых служб.

Удаленный доступ

Удаленным доступом (DialUP) называют получение доступа к разрешенным папкам компьютера с помощью модема или виртуального частного соединения (VPN). Поскольку VPN является не таким уж частым гостем на домашних компьютерах пользователей, то в данной главе будет рассказано только о настройках модема.

HKEY_CURRENT_USER\Software\Microsoft\RAS Phonebook

Эта ветвь реестра содержит параметры, используемые при дозвоне к провайдеру с помощью модема. Среди этих параметров наиболее интересны следующие.

- PopupOnTopWhenRedialing — определяет, будет ли выводиться сообщение об ошибке при неудачном дозвоне к провайдеру. Если значение этого параметра равно 0, то сообщение выводиться не будет. По умолчанию значение этого параметра равно 1.

- `OperatorDial` — указывает, нужно ли самостоятельно набирать номер провайдера или это автоматически сделает компьютер. Если значение равно `1`, то при удаленном соединении компьютер предложит вам набрать номер провайдера самому или попросить об этом оператора.

Параметры имеют `DWORD`-тип.

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class\ {4D36E96D-E325-11CE-BFC1-08002BE10318}\0000

Эта ветвь реестра содержит настройки первого модема (если на компьютере установлено несколько модемов, то ветвь реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class\{4D36E96D-E325-11CE-BFC1-08002BE10318}` будет включать в себя несколько подразделов) как класса оборудования. При этом раздел `{4D36E96D-E325-11CE-BFC1-08002BE10318}` может иметь и другой GUID. В этом случае класс модемов необходимо искать по параметру строкового типа `Class`, расположенному в ветви `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class\{GUID-íîìåð êëàññà}`. Для класса модемов данный параметр должен содержать значение `Modem`. Соответствие стандартных классов устройствам, в них описанным, можно просмотреть в файле `certclas.inf`, расположенном в каталоге `%systemroot%\inf`.

Ветвь реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class\{4D36E96D-E325-11CE-BFC1-08002BE10318}\0000` может хранить следующие параметры.

- `AttachedTo` — этот параметр строкового типа определяет номер COM-порта, по которому будет работать данный модем. Например, значение параметра может быть равно `COM1`.
- `Blind_Off` — параметр строкового типа, указывает командную строку, используемую для обнаружения сигнала в линии. Например, значение параметра может быть равно `X5`. Параметр может находиться либо в описываемой ветви, либо в ветви `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class\{4D36E96D-E325-11CE-BFC1-08002BE10318}\0000\Settings`.
- `Blind_On` — этот параметр строкового типа определяет командную строку, используемую для обнаружения сигнала в линии. Например, значение параметра может быть равно `X3`. Параметр может находиться либо в описываемой ветви, либо в ветви `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class\{4D36E96D-E325-11CE-BFC1-08002BE10318}\0000\Settings`.
- `InactivityScale` — параметр `BINARY`-типа, указывает коэффициент, используемый для вычисления тайм-аута подключения.

ПРИМЕЧАНИЕ

Содержимое приведенной выше ветви реестра более подробно описано в базе данных реестра, содержащейся на компакт-диске, поставляемом с книгой.

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\ RasMan\Parameters

Эта ветвь реестра хранит настройки службы удаленного доступа. В ней могут находиться следующие параметры DWORD-типа.

- DisableSavePassword — определяет, будет ли доступен флажок Сохранять имя пользователя и пароль при удаленных соединениях. Если значение равно 1, то флажок будет недоступен. По умолчанию значение равно 0.
- NumberOfRings — указывает количество дозвонов, которое будет выполнять компьютер при соединении с провайдером.
- LimitSimultaneousIncomingCalls — определяет максимальное возможное количество входящих звонков на удаленное подключение. Если значение равно 0, то данному компьютеру будет запрещено принимать входящие вызовы на удаленное подключение. По умолчанию значение равно 3.
- LimitSimultaneousOutgoingCalls — указывает максимальное возможное количество исходящих звонков на удаленное подключение. Если значение равно 0, то данному компьютеру будет запрещено инициировать (отправлять вызовы) на удаленное подключение. По умолчанию значение равно 4.

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\ RasMan\PPP

Эта ветвь реестра хранит настройки протокола PPP, используемого службой удаленного доступа для организации подключения с помощью модема. В ветви может находиться параметр SecurePVN. Значение этого параметра DWORD-типа определяет, будет ли использоваться протокол MS-CHAPv2 (вторая версия этого протокола считается более защищенной, чем первая) при соединении с применением протокола PPTP. По умолчанию значение равно 0, то есть протокол MS-CHAPv2 использоваться не будет.

Но, кроме параметров, раздел содержит и вложенные подразделы. Например, в раздел PPP вложен подраздел EAP. Он определяет настройки протокола обеспечения безопасности EAP и может включать в себя другие вложенные подразделы.

- 13 — настройки из данной ветви реестра применяются, если в списке, открываемом при установке переключателя в положение Протокол расширенной проверки подлинности (EAP) (данный переключатель расположен в диалоговом окне Дополнительные параметры шифрования, открываемом при нажатии кнопки Параметры на вкладке Безопасность свойств модема), выбран элемент Смарт-карта или иной сертификат.
- 25 — настройки применяются, если в списке, открываемом при установке переключателя в положение Протокол расширенной проверки подлинности (EAP), выбран элемент Защищенные EAP (PEAP).
- 26 — настройки из данной ветви реестра применяются, если в списке, открываемом при установке переключателя в положение Протокол расширенной проверки подлинности (EAP), выбран элемент Безопасный пароль (EAP-MSCHAP v2).

- 4 — настройки применяются, если в списке, открываемом при установке переключателя в положение Протокол расширенной проверки подлинности (EAP), выбран элемент MD5-задача.

Все эти подразделы могут содержать следующие параметры.

- FriendlyName — этот параметр DWORD-типа указывает название, отображаемое в списке, открываемом при установке переключателя в положение Протокол расширенной проверки подлинности (EAP).
- RolesSupported — параметр DWORD-типа, определяет поддерживаемую роль данного способа аутентификации. Параметр имеет одну интересную особенность — с его помощью можно исключить из списка, открываемого при установке переключателя в положение Протокол расширенной проверки подлинности (EAP), возможность использования соответствующего способа аутентификации (то есть скрыть соответствующий элемент списка). Например, чтобы из списка скрыть элемент Смарт-карта или иной сертификат, достаточно данному параметру, расположенному в подразделе 13, присвоить значение 1 (по умолчанию значение параметра равно 2). Если же присвоить значение 9 (по умолчанию значение параметра равно a) данному параметру из подраздела 4, то из списка исчезнет элемент MD5-задача.

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\RemoteAccess\Parameters

Эта ветвь реестра принадлежит службе маршрутизации и удаленного доступа. В ней могут находиться следующие параметры DWORD-типа.

- AutoDisconnect — определяет время задержки в секундах, после которого неактивное соединение с удаленным сервером будет отключено.
- CallbackTime — указывает задержку перед инициализацией отзыва. Он может принимать значения в диапазоне от 2 до 12.

Но, кроме параметров, описываемая ветвь реестра может содержать и вложенные подразделы. Например, к таким подразделам можно отнести AccountLockout, параметры которого определяют поведение системы в случае неверного ввода пароля пользователя при попытке удаленного доступа. Данный подраздел может включать в себя следующие параметры DWORD-типа.

- MaxDenials — определяет количество попыток неверного ввода пароля при удаленном доступе, после которого учетная запись данного пользователя будет заблокирована.
- ResetTime (mins) — указывает интервал времени (в минутах), на который будет заблокирована учетная запись пользователя, превысившего количество вводов неверного пароля.

Стек протоколов TCP/IP

Стек протоколов TCP/IP является основным и единственным способом взаимодействия конечного компьютера с глобальной сетью Интернет, а также основным

стеком взаимодействия с другими компьютерами сети. Именно поэтому было решено рассказать в этой главе и о некоторых настройках стека TCP/IP. Все настройки стека TCP/IP хранятся в ветви реестра HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters. Она может содержать следующие параметры DWORD-типа.

- ArpCacheLife — указывает время жизни записей в ARP-кэше.
- DefaultTTL — определяет время жизни пакетов (TTL), указываемое в заголовках пакетов. Время жизни пакета — это количество маршрутизаторов, через которые может пройти пакет, после чего он станет считаться утерянным и будет уничтожен. Каждый маршрутизатор, передающий пакет, уменьшает его время жизни на 1. Когда время жизни становится равным нулю, следующий принимающий пакет маршрутизатор уничтожает его.

  По умолчанию значение равно 0õ00000080, но параметр может принимать значения от 0õ00000001 до 0õ00000100.
- DisableTaskOffload — определяет, будет ли при взаимодействии с сетевым компьютером использоваться дополнительный сопроцессор сетевой карты. Использование дополнительного сопроцессора разгружает работу процессора, установленного на материнской плате компьютера, но не все сетевые карты имеют в своем составе дополнительные сопроцессоры.

  По умолчанию значение равно 1, то есть, даже если сетевая карта имеет дополнительный сопроцессор, он будет отключен. Чтобы включить работу сопроцессора, необходимо присвоить этому параметру значение 0.
- EnablePMTUBHDetect — указывает, будет ли при передаче пакетов выполняться поиск маршрутизаторов типа «черная дыра». По умолчанию поиск данных маршрутизаторов отключен, то есть параметр имеет значение 0. Установка значения параметра равным 1 приводит к дополнительной затрате времени на стадии установки соединения.

ПРИМЕЧАНИЕ

Маршрутизаторы типа «черная дыра» не возвращают пакеты типа ICMP Destination Unreachable в случае, если им нужно фрагментировать пакет с установленным флагом, запрещающим фрагментацию. Именно такие пакеты будут использоваться TCP для обнаружения MTU пути. Если после нескольких попыток передачи пакетов с запрещенной дефрагментацией не было получено ответа, то считается, что маршрутизатор имеет тип «черная дыра». Если подтверждение будет получено, то MSS будет уменьшено и флаг, запрещающий дефрагментацию, будет установлен для всех последующих пакетов.

- EnablePMTUDiscovery — указывает, будет ли TCP перед отправкой пакетов определять максимальный размер пакета (MTU), который не будет фрагментироваться во время передачи к сетевому компьютеру. Если значение равно 0, то все передаваемые пакеты будут иметь MTU, равный 576 байт, а определение максимального размера пакета, способного дойти до сетевого компьютера без фрагментации, вестись не будет. Если же значение равно 1, то перед отправкой пакетов

TCP будет определять MTU пакета, способного дойти до сетевого компьютера без фрагментации, и полученный MTU использовать как размер пакетов, которые он будет отправлять (чтобы они не фрагментировались, ведь эта операция занимает дополнительное время). По умолчанию значение параметра равно `1`.

- `ForwardBufferMemory` — указывает размер буфера, который используется IP для хранения данных пакета в очереди пакетов маршрутизатора (когда данный буфер заполняется, маршрутизатор в произвольном порядке удаляет пакеты из буфера). Значение должно быть кратно 256 байт, так как буферы данных очереди пакетов по умолчанию имеют размер в 256 байт. Параметр может принимать значение от величины MTU пакета до `0xFFFFFFFF`. По умолчанию значение равно `74240`.

ПРИМЕЧАНИЕ

Заголовки IP-пакетов хранятся в отдельном буфере.

- `KeepAliveInterval` — определяет интервал отправки пакетов проверки активности до тех пор, пока не будет получен ответ хотя бы на один пакет. Если ответный пакет получен, то отправка пакетов активности прекращается до тех пор, пока не истечет интервал времени, задаваемый `DWORD`-параметром `KeepAliveTime`. По истечении данного времени отправка пакетов активности опять начинается.

  Если же после отправки количества пакетов активности, заданного в `DWORD`-параметре `TcpMaxDataRetransmissions`, ни на один из них не было получено ответа, то данное соединение считается неактивным и разрывается. По умолчанию значение параметра `KeepAliveInterval` равно `1000`. Допустимы значения в диапазоне от `1` до `0xFFFFFFFF`.

- `KeepAliveTime` — указывает интервал ожидания перед началом отправки пакетов активности (Keep Alive Packet), на которые удаленный компьютер должен ответить, иначе будет разорвано соединение. По умолчанию пакеты активности не отправляются, но их отправку может инициировать пользовательское приложение. По умолчанию значение данного параметра равно `7200000` (два часа).
- `MTU` — определяет максимальный размер передаваемого пакета данных. По умолчанию значение равно `0õ000005DÑ`.
- `NumForwardPackets` — указывает количество заголовков IP-пакетов, которые могут находиться в очереди пакетов маршрутизатора. Если реальное количество IP-заголовков превышает значение данного параметра, то маршрутизатор в случайном порядке начинает отбрасывать пакеты из очереди. Значение может находиться в диапазоне от `1` до `0xFFFFFFFE`.

ПРИМЕЧАНИЕ

Значение этого параметра должно быть не меньше значения параметра ForwardBufferMemory, деленного на максимальный размер данных IP в сети, подключенной к этому маршрутизатору. При этом значение данного параметра должно быть не больше значения параметра ForwardBufferMemory, деленного на 256.

- SackOpts — определяет, включена ли возможность SACK (впервые эта возможность была реализована в Windows XP). По умолчанию значение равно 0, то есть данная возможность отключена. Если же значение равно 1, то при потере пакета отправитель может передать лишь потерянный пакет, а не все пакеты сообщения, пакет которого был потерян.
- SynAttackProtect — указывает, будет ли на компьютере задействована встроенная защита от DOS-атаки SYN-переполнением. Если значение равно 0, то защита будет отключена. Например, если установлен прокси-сервер, то на клиентских машинах, как правило, нет нужды в такой защите. Если значение равно 1, то будет включена стандартная возможность защиты от SYN-атак. Если же значение параметра равно 2, то, кроме стандартной защиты от SYN-атак, используются дополнительные меры защиты: операционная система будет выполнять обращения к драйверу AFD (поддержка Windows Sockets) только в случае установки полного соединения.
- Tcp1323Opts — определяет, будут ли при сетевом соединении использоваться более широкие окна (под шириной окна понимается количество пакетов, которые может передать отправитель до получения подтверждения приема пакетов от получателя) передачи пакетов. Если значение равно 3, то будут использоваться более широкие окна, что может повысить скорость передачи для высокоскоростных сетевых соединений. По умолчанию значение параметра равно 0.
- TcpMaxDataRetransmissions — указывает количество попыток передачи данных, после которых (если они были неуспешны) соединение будет окончательно разорвано. Значение может находиться в пределах от 0 до 0xFFFFFFFF. По умолчанию оно равно 15.
- TcpMaxHalfOpen — определяет максимальное возможное количество одновременных полуоткрытых соединений, которое поддерживает TCP.
- TcpNumConnections — указывает максимальное возможное количество одновременных соединений, которое поддерживает TCP. Значение может находиться в пределах от 0 до 0xFFFFFE. По умолчанию оно равно 0xFFFFFE.
- TCPWindowSize — определяет ширину окна для приема TCP (то есть количество байт, которые могут быть переданы отправителем без приема квитанции о подтверждении получения адресатом). Значение может находиться в пределах от 0 до 65535. По умолчанию оно равно 0õFFFF (65535).

ПРИМЕЧАНИЕ

Для большей эффективности работы сети размер окна, задаваемый параметром Tcpwindowsize, должен быть кратен MSS.

Рабочая станция

Рабочая станция — это служба на компьютере пользователя, с помощью которой реализуется поддержка сетевых подключений и связи между компьютерами. Без использования данной службы невозможно осуществление доступа к другим

компьютерам. При этом настройки службы оказывают влияние на скорость подключения к сетевым компьютерам. Параметры реестра, относящиеся к настройке рабочей станции, расположены в ветви реестра HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\lanmanworkstation\parameters.

Буфер сетевых операций

Буфер сетевых операций используется для хранения сетевых команд и потоков. Увеличение размера буфера может повысить скорость передачи данных. Но следует учитывать, что память для буфера сетевых операций резервируется за счет оперативной памяти, поэтому, чем больше будет буфер сетевых операций, тем меньше окажется оперативной памяти.

Для настройки буфера сетевых операций используются следующие параметры DWORD-типа.

- MaxCmds — определяет количество команд, которые могут находиться в буфере для сетевых операций в одну единицу времени. Возможные значения параметра находятся в диапазоне от 0 до 255. По умолчанию значение равно 15.
- MaxThreads — указывает количество потоков, которые могут находиться в буфере для сетевых операций в одну единицу времени. Возможные значения параметра находятся в диапазоне от 0 до 255. По умолчанию значение равно 15.
- MaxCollectionCount — определяет размер буфера, который будет использоваться для записи через именованные каналы, работающие в символьном режиме. Значения параметра находятся в диапазоне от 0 до 65535. По умолчанию значение равно 16.

Другие способы оптимизации Windows

Напоследок рассмотрим несколько способов оптимизации Windows, направленных на настройку оболочки операционной системы, а также на очистку содержимого файловой системы.

Интерфейс пользователя

Настройки интерфейса пользователя, применяемые по умолчанию в операционной системе Windows XP, не рассчитаны на компьютеры малой мощности или с небольшим объемом оперативной памяти. Поэтому при входе пользователя включены многие эффекты, которые замедляют работу компьютера, да и вообще могут быть непривычны пользователям, работавшим ранее в операционных системах Windows 2000 или более ранних версиях Windows. Рассмотрим некоторые из этих настроек. При этом в основном будет приводиться описание диалога, в котором можно настроить данный параметр, и только в случае отсутствия подобного диалога будет рассматриваться ветвь реестра и параметр, отвечающий за настройку эффекта.

Новый диалог входа пользователя в систему

В Windows XP появился новый диалог для входа пользователя в систему, применяемый по умолчанию. Если раньше в операционной системе Windows 2000 вас встречало компактное окно для ввода имени пользователя и пароля, то теперь вас ждет красиво оформленный диалог, в котором можно выбрать учетную запись пользователя, от имени которого необходимо выполнить вход в систему, а потом ввести пароль для данной учетной записи.

Естественно, что за красоту нужно расплачиваться, и в данном случае пользователь расплачивается сниженной скоростью входа в систему, а также отображения самого диалога ввода пароля. Если скорость входа в систему для вас критична, то можно изменить способ входа, вернув на место стандартный диалог Windows 2000. Для этого необходимо запустить апплет `nusrmgr.cpl`, в котором выбрать гиперссылку Изменение входа пользователей в систему, а потом снять флажок Использовать страницу приветствия. Можно это сделать и проще — просто присвоить DWORD-параметру `LogonType` значение `0`. Параметр расположен в ветви системного реестра `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon`.

Предыдущий совет немного повысит скорость входа в систему, но все равно потребует от вас ввода имени пользователя и пароля. Если вы не боитесь несанкционированного проникновения в систему, то можно указать имя пользователя и пароль, которые будут автоматически учитываться при входе в систему. На страницах этой книги уже было описано два способа выполнения данной процедуры. Первый использовал команду `rundll32`, а второй — параметры реестра, которые необходимо изменить. Сейчас же применим третий способ — введем команду `control USERPASSWORDS2`. После этого отобразится знакомый диалог, в котором нужно снять флажок Требовать ввод имени пользователя и пароля. После того как вы нажмете кнопку ОК или Применить, система попросит вас ввести имя пользователя, с правами которого будет выполняться вход, и пароль его учетной записи.

Новый вид меню Пуск

Это еще одна часть оболочки, которая может не нравиться пользователям, привыкшим к стандартному виду меню Пуск для Windows 2000. Как правило, новый вид меню Пуск отображается немного дольше обычного, а со временем вообще превращается в кашу из ссылок на различные программы (как, в принципе, и стандартное меню Пуск).

Чтобы переключиться на стандартный вид меню Пуск, необходимо вызвать диалог Свойства панели задач и меню "Пуск" либо выбрав соответствующую команду пункта Настройка меню Пуск, либо воспользовавшись командой `rundll32 shell32.dll, Options_RunDLL 1`. После отображения диалога необходимо перейти на вкладку Меню "Пуск" и установить переключатель в положение Классическое меню "Пуск".

Вопрос превращения меню Пуск в кашу также можно решить. Наиболее простым способом его решения будет очистка меню Пуск от ненужных ярлыков, а после этого

редактирование параметров доступа к каталогам `%userprofile%\Ãëàâíîå ìåíþ` и `%systemdrive%\Documents and Settings\All Users\Ãëàâíîå ìåíþ`. При этом желательно запретить запись в данные каталоги не только своей, но и системной учетной записи (оставив только доступ на чтение). Этим вы добьетесь сразу двух целей: во-первых, устанавливаемые программы не смогут добавить свои ярлыки в меню Пуск, а во-вторых, они не смогут воспользоваться каталогом Автозагрузка для своего запуска при каждом входе пользователя в систему.

Автозапуск программ при входе пользователя в систему

Раз уж была затронута тема запрета запуска программ с помощью каталога Автозагрузка, поговорим о другом методе запуска программ — посредством реестра Windows. В реестре Windows XP существует много ветвей, из которых программа может быть автоматически запущена, но основной ветвью, используемой для этого, является `HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Run` (а также ветвь корневого раздела `HKEY_LOCAL_MACHINE`). Поэтому для этих ветвей также желательно установить только доступ на чтение, как для своей учетной записи, так и для учетной записи системы. Этим вы решите два вопроса. Во-первых, сделаете невозможным установку автоматического запуска новых программ, а во-вторых, повысите общий уровень защиты от вирусов и других хакерских программ, которые запускаются при входе пользователя в систему, так как они чаще всего используют для своего запуска именно эти ветви реестра.

ПРИМЕЧАНИЕ

Другие ветви для автоматического запуска программ при входе пользователя в систему будут описаны в конце данной главы.

При упоминании об автоматическом запуске программ следует также сказать о такой программе, как `msconfig.exe`. Она содержит две полезные вкладки: Службы и Автозагрузка. С помощью вкладки Службы можно запретить автоматический запуск определенных служб, установленных на компьютере. С помощью вкладки Автозагрузка можно запретить автоматический запуск программ. При этом на данной вкладке находится список программ, запускаемых как с помощью приведенной выше ветви реестра, так и с помощью каталога Автозагрузка. Только не следует переусердствовать при отключении программ. Например, следующие программы могут вам понадобиться:

- `mobsync` — используется для синхронизации автономных файлов при входе пользователя в систему (если эта функция не используется, можно отключить);
- `ctfmon` — является программой для отображения языковой панели (и если вы используете языковую панель, то данную программу отключать не нужно).

Если вы запретили автоматический запуск какой-либо программы, то после закрытия программы `msconfig.exe` система предложит вам перезагрузить компьютер.

Оптимизация оболочки с помощью диалоговых окон

Теперь рассмотрим некоторые параметры стандартных диалогов Windows XP, редактирование которых может повысить общую скорость работы того или иного компонента системы.

Во-первых, диалоговое окно Свойства: Экран. На вкладке Оформление этого окна есть кнопка Эффекты, после нажатия которой отобразится одноименный диалог. Этот диалог содержит следующие флажки.

- Применять следующий переходный эффект для меню и подсказок — если вы хотите, чтобы меню открывались немного быстрее, то лучше этот флажок снять.
- Отображать тени, отбрасываемые меню — можно также снять. Во-первых, это повысит скорость открытия меню, а во-вторых, не всем могут нравиться стандартные тени меню Windows XP.
- Отображать содержимое окна при перетаскивании — если снять этот флажок, то при перетаскивании окон их содержимое будет скрываться (останется только рамка окна). С одной стороны, это повышает скорость работы с окнами Проводника, в которых расположено очень много папок и файлов. Но, с другой стороны, к такому способу перетаскивания нужно привыкнуть, ведь скрываться будет содержимое не только окон Проводника, но и, например, окон графического редактора Adobe Photoshop, что может быть неприемлемо.

Во-вторых, диалоговое окно Свойства системы. На вкладке Дополнительно этого диалога нужно нажать кнопку Параметры, расположенную в области Быстродействие. После этого отобразится диалог Параметры быстродействия, содержащий список различных настроек.

- Анимация окон при свертывании и развертывании — снятие флажка повысит скорость сворачивания и разворачивания окон. Если компьютер для вас не элемент красоты, а рабочий инструмент, то лучше снять.
- Использование типичных задач для папок — снятие этого флажка приведет к скрытию области типичных задач, расположенной слева в окнах Проводника. Данная область призвана облегчить работу с файловой системой Windows XP, храня наиболее часто используемые функции как самого окна, так и отдельных выделенных файлов в нем. В основном данное окно дублирует различные команды контекстного меню, поэтому без его помощи можно легко обойтись. Хотя это дело привычки.

Данный диалог содержит многие другие настройки, но они мало влияют на скорость работы оболочки Windows XP.

Оптимизация оболочки с помощью параметров реестра

Рассмотрим также несколько параметров реестра, влияющих на оптимизацию оболочки, но не имеющих способа изменения с помощью стандартных диалогов Windows.

- `MenuShowDelay` — данный параметр строкового типа уже был рассмотрен ранее (расположен в ветви реестра `HKEY_CURRENT_USER\Control Panel\Desktop`).

Он определяет задержку в миллисекундах перед отображением меню. Естественно, что чем меньше задержка, тем быстрее будут открываться меню, хотя здесь лучше не переусердствовать. Меньше значения 100 лучше не опускаться.

- UserPreferencesMask — этот параметр REG_BINARY-типа расположен в ветви реестра HKEY_CURRENT_USER\Control Panel\Desktop. Он является битовой маской, один бит которой хотелось бы рассмотреть. Это бит 0õ00200000. Если данный бит установлен, то будет использоваться альтернативное контекстное меню, отображение которого, как правило, выполняется быстрее.

ПРИМЕЧАНИЕ

После установки бита меню программ примет коричневатый оттенок.

Файловая система

Теперь скажем несколько слов о содержимом файловой системы Windows. Как правило, здесь также есть над чем поработать, особенно если места на жестком диске мало. Автор лишь приведет определенные пути к каталогам и ветви реестра, а вы сами решайте, нужны ли они вам.

- %systemroot%\Installer — является скрытым и хранит копии пакетов установщика Windows, которые вы когда-либо запускали. Он может понадобиться при повреждении файлов программ. К тому же, как правило, если вы когда-то удалили программу, а сейчас вам необходимо ее установить, но пакета установщика данной программы у вас нет, можно попробовать поискать его среди содержимого данного каталога. Если же места на диске мало, то можно удалить этот каталог (сначала рекомендуется просто переименовать его, перезагрузиться и поработать с программами, если ни одна программа не требует для своего открытия пакета установщика и ведет себя как обычно, то можно удалить этот каталог).
- %systemroot%\$íàçâàíèå îáíîâëåíèÿ$ — каталоги такого формата содержат файлы операционной системы, которые были заменены при установке обновления или заплаты для компонентов операционной системы. Они необходимы для реализации возможности возврата к предыдущему состоянию системы, если после установки обновления система ведет себя некорректно.
- %systemroot%\LastGood — еще один каталог файловой системы Windows XP, который иногда можно встретить. Он содержит копии системных файлов, которые гарантированно работают. Если система работает стабильно, то можно удалить этот каталог.
- %systemroot%\system32\dllcache — является скрытым и хранит копии системных файлов, предназначенные для замены используемых системных файлов в случае их повреждения или незаконного изменения. По умолчанию он занимает очень много места (около 400 Мбайт), хотя размер, отводимый для него, можно изменить с помощью DWORD-параметра SfcQuota, расположенного в ветви реестра HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon. Но автор все-таки не советовал бы изменять

размер этого каталога или удалять его (он все равно будет восстановлен, если вы его удалите). Единственное, что можно посоветовать, это использование одного каталога для всех операционных систем, установленных на компьютере. Если у вас на компьютере установлено две версии Windows XP (желательно одинаковых), то можно заставить эти операционные системы использовать единственный каталог dllcache. Путь, по которому располагается этот каталог, хранится в реестре. Для этого предназначен параметр строкового типа SFCDllCacheDir, расположенный в ветви реестра HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Microsoft\Windows NT\Windows File Protection (по умолчанию не существует). Измените в этом параметре путь к каталогу (например, на d:\recent\dllcache), а потом переместите сам каталог по указанному в параметре пути (в данном случае нужно переместить в каталог d:\recent). После этого нужно перезагрузить компьютер, и если после этого не было создано новой папки dllcache в каталоге %systemroot%\system32, то все хорошо. Аналогично нужно изменить параметр в реестре другой операционной системы, чтобы он ссылался на ту же папку dllcache, а старую папку второй операционной системы удалить.

- %systemroot%\Driver Cache — содержит архив всех драйверов, поставляемых на установочном диске операционной системы. Он может использоваться при поиске подходящего драйвера для нового устройства вместо установочного диска Windows XP. Хотя если вы готовы при каждом поиске нового драйвера доставать установочный диск Windows вместо того, чтобы выполнить поиск в этом каталоге, то его можно удалить. Если же у вас на компьютере установлены две одинаковые версии Windows XP, то можно изменить путь к этому каталогу, чтобы операционными системами использовался общий каталог, а не отдельный для каждой системы. Путь к данному каталогу хранится в параметре строкового типа DriverCachePath, расположенном в ветви реестра HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Setup.
- %systemroot%\Temp — предназначен для хранения временных файлов, необходимых при установке программ. Как правило, именно сюда система распаковывает файлы и установочные пакеты программ перед тем, как начать их установку. Поэтому иногда рекомендуется просматривать содержимое данного каталога и удалять уже ненужные файлы, ведь он, как и рассмотренный каталог %systemroot%\Installer, может хранить все пакеты установщика Windows, устанавливаемые на вашем компьютере, а также другие файлы, используемые при установке.
- %systemroot%\Minidump — содержит файлы малого дампа памяти, создаваемые при аварийной остановке системы. Каждый из этих файлов занимает 92 Кбайт, хотя если «синий экран» для вас не в новинку, то через несколько месяцев может собраться неплохой список файлов.

Это далеко не весь список каталогов, содержимое которых при нехватке места можно попробовать удалить. Если у вас намечена генеральная чистка жесткого диска, то можно заглянуть в ветвь реестра HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\VolumeCaches. Она включает в себя список разделов, каждый из которых определяет список файлов или папок,

использовавшихся при установке различных компонентов Windows или программ и теперь больше не нужных. Другими словами, теоретически их можно удалить, хотя система почему-то этого не делает. Разделы данной ветви могут содержать следующие строковые параметры:

- `Description` — описывает, когда и кем использовались данные файлы и папки и почему они больше не нужны;
- `FileList` — включает в себя список файлов, которые можно удалить;
- `Folder` — указывает путь к каталогу, в котором хранятся эти файлы;
- `CleanupString` — может содержать команду, с помощью которой можно автоматически удалить описываемые временные файлы.

Глава 8

Ветвь реестра HKEY_LOCAL_MACHINE\SYSTEM

- **Раздел ControlSetNNN**
- **Раздел Select**
- **Настройки служб**
- **Потенциально опасные ветви и параметры реестра**

Ветвь реестра HKEY_LOCAL_MACHINE\SYSTEM является наиболее важной для загрузки системы. Если она будет повреждена, то с большой долей вероятности вы уже не сможете войти в систему. Поэтому, наверное, стоит несколько слов сказать и о структуре этой ветви.

Структура ветви большей частью статична. Иными словами, независимо от того, как будет называться новый раздел, добавляемый к содержимому разделов ветви, параметры, которые он должен включать в себя, предопределены программистами Microsoft. Вообще, эта ветвь реестра предназначена для хранения сведений обо всех драйверах, службах и сервисах, установленных в системе. Но, кроме этого, ветвь содержит критически важные сведения настройки самой системы. Пример таких сведений можно найти в последней части книги.

Ветвь реестра HKEY_LOCAL_MACHINE\SYSTEM может включать в себя следующие разделы.

- CurrentControlSet и разделы ControlSetNNN — как раз и определяют все сведения о сервисах и службах, установленных в вашей системе, способ и последовательность их запуска, а также различные настройки сетевых компонентов и самой операционной системы.
- MountedDevices — указывает настройки монтирования логических дисков вашей системы.
- Select — определяет ветви ControlSetNNN и способ их использования, но о нем мы поговорим чуть позже.
- Setup — указывает настройки установки Windows, а также может использоваться программой sysprep для своего запуска при следующей перезагрузке.
- WPA — содержит сведения об активационных ключах, доступных вашей операционной системе.

Раздел ControlSetNNN

Теперь подробнее поговорим о самых важных разделах ветви системного реестра HKEY_LOCAL_MACHINE\SYSTEM. Первыми из них будут разделы формата ControlSetNNN и раздел CurrentControlSet. Об их важности говорит уже то, что, хотя в системе может содержаться несколько разделов формата ControlSetNNN (вместо NNN указывается номер раздела, например ControlSet001, ControlSet002 или ControlSet003), все они хранят практически одинаковую информацию. И это не избыточность. Программисты Microsoft приняли решение специально использовать несколько копий разделов, содержащих критически важную информацию, чтобы в случае повреждения одного из них система могла загрузиться с помощью настроек из другого раздела.

Каждый из разделов формата ControlSetNNN используется как страховочный. При этом за двумя из этих разделов всегда зарезервирован свой вид загрузки операционной системы — один из разделов используется для обычной загрузки, а второй применяется при выборе пользователем из списка альтернативных видов

загрузки команды Загрузка последней удачной конфигурации. Второй раздел используется в том случае, если систему не удалось загрузить с помощью первого раздела.

Раздел CurrentControlSet на самом деле не является физически существующим в реестре, его содержимое — это лишь ссылка на тот раздел ControlSetNNN, который был загружен в текущий момент.

Для понятия принципа работы данных разделов системы необходимо знать этапы загрузки операционной системы Windows и то, что на этих этапах происходит. Мы же не будем углубляться так далеко, а перечислим лишь несколько фактов, которые помогут в понимании сути рассматриваемых разделов. После сбора информации о конфигурации компьютера и выбора самой загружаемой системы (если используется мультизагрузка) происходит попытка считывания ветви реестра HKEY_LOCAL_MACHINE\SYSTEM\ControlSetNNN, которая в данный момент используется для обычной загрузки. Если на этапе считывания или попытки запуска какого-нибудь драйвера, указанного в ветви реестра HKEY_LOCAL_MACHINE\SYSTEM\ControlSetNNN, происходит серьезный сбой, то ветвь системного реестра ControlSetNNN помечается как испорченная. После этого начинается новая перезагрузка компьютера, в процессе которой уже используется ветвь реестра HKEY_LOCAL_MACHINE\SYSTEM\ControlSetNNN, помеченная как удачная при предыдущем удачном запуске операционной системы. Если же системе удается загрузиться при помощи ветви HKEY_LOCAL_MACHINE\SYSTEM\ControlSetNNN и при этом в системе удачно зарегистрировался хотя бы один пользователь, то данная ветвь ControlSetNNN считается корректной, и теперь именно с ее помощью вы будете загружаться при выборе команды Загрузка последней удачной конфигурации. После завершения работы компьютера все занесенные вами в текущий сеанс работы сведения помещаются в используемый при загрузке системы раздел ControlSetNNN. Остальные же разделы остаются без изменений.

Раздел Select

Но как же система узнает, какой из разделов ControlSetNNN необходимо использовать при обычной загрузке, какой нужно применять при загрузке последней удачной конфигурации, а какой вообще является испорченным? Именно для этих целей и предназначен раздел Select. Он содержит параметры DWORD-типа, каждый из которых определяет номер раздела ControlSetNNN и ту метку, которая была ему присвоена во время последнего удачного входа в систему. Рассмотрим назначение каждого из параметров, описанных в разделе Select.

- Default — определяет, какая копия раздела ControlSetNNN будет загружена при обычной загрузке системы. Например, если значение данного параметра равно 2, то при обычной загрузке системы раздел CurrentControlSet будет ссылкой на содержимое раздела ControlSet002.
- Current — указывает номер текущей копии раздела ControlSetNNN, который использовался для загрузки системы и на который ссылается раздел CurrentControlSet.

- `LastKnownGood` — определяет номер копии раздела `ControlSetNNN`, которая будет использоваться для загрузки и построения содержимого раздела `CurrentControlSet` при использовании команды меню альтернативной загрузки Загрузка последней удачной конфигурации.
- `Failed` — указывает раздел `ControlSetNNN`, при предыдущей загрузке которого произошел какой-то серьезный сбой и загрузка с его помощью была прервана.

Настройки служб

После рассмотрения назначения разделов формата `ControlSetNNN` вы знаете, что они предназначены для хранения настроек запускаемых системой служб. Но как эти настройки хранятся в реестре? Именно этому вопросу и посвящен данный раздел.

Все настройки запуска служб хранятся в ветви реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services`. Она содержит список разделов, каждый из которых определяет описания одной службы или сервиса. Названия данных разделов, в принципе, не имеют значения (но если для службы не существует параметра `DisplayName`, то для ее идентификации будет использоваться название раздела, в котором она описывается). Значение имеют те параметры, которые описаны в соответствующем разделе. К таким параметрам можно отнести приведенные ниже.

- `Group` — параметр имеет тип `REG_SZ` и определяет группу, к которой относится служба. Именно от группы зависит, в какой момент будет запущена служба — сначала запускаются все службы одной группы, потом все службы другой и т. д. Саму же последовательность, в которой запускаются группы служб, можно просмотреть в `REG_MULTI_SZ`-параметре `List`, расположенном в ветви реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\ServiceGroupOrder`.
- `DependOnGroup` — этот параметр `REG_MULTI_SZ`-типа определяет группы, которые должны быть запущены перед запуском данной службы. В контексте оснастки `services.msc`, которая описывает все службы, установленные на компьютере, данный параметр определяет содержание вкладки Зависимости диалога Свойства для данной службы.
- `DependOnService` — параметр `REG_MULTI_SZ`-типа, определяет сервисы, которые должны быть запущены перед запуском данной службы. Значения этого параметра отображаются на вкладке Зависимости диалога Свойства, вызываемого двойным щелчком левой кнопкой мыши на строке, определяющей данную службу в оснастке `services.msc`.
- `DisplayName` — этот параметр строкового типа определяет строку названия службы, которая как раз и будет идентифицировать службу в оснастке `services.msc` (данная строка будет отображаться в поле Имя оснастки `services.msc`).

- Description — параметр строкового типа, определяет строку описания для соответствующей службы. Строка будет отображаться в поле Описание диалога Свойства для данной службы.
- ObjectName — этот параметр строкового типа определяет учетную запись, с правами которой будет запускаться служба. Если его значение равно LocalSystem, то вход будет выполняться с правами данной учетной записи (эта запись пришла на смену записи System, определяющей права системы, и содержит меньше прав, чем сама учетная запись System). Если же значение этого параметра равно NT Authority\NetworkService, то вход будет выполнен от имени сетевой службы (аналогично учетной записи LocalSystem, данная учетная запись имеет меньше прав, чем учетная запись System). Если же вам необходимо предоставить службе вход от имени учетной записи определенного пользователя данного компьютера, то параметру ObjectName в качестве значения нужно присвоить строку формата .\ëîãèí ïîëüçîâàòåëÿ.
- ErrorControl — параметр DWORD-типа, определяет поведение системы при возникновении ошибок в работе службы и может принимать такие значения:
  - 0 — игнорировать ошибку;
  - 1 — предупреждать пользователя об ошибке;
  - 2 — перезагрузить компьютер.
- ImagePath — этот параметр строкового типа определяет путь к файлу службы, который и будет запускаться системой. В оснастке services.msc параметр определяет содержимое поля Исполняемый файл диалога Свойства для соответствующей службы (это поле позволяет лишь просмотреть путь к файлу службы, но не отредактировать его).
- Start — параметр DWORD-типа, определяет момент загрузки системы, в который будет запущена данная служба. Он может принимать следующие значения:
  - 0 — служба будет запускаться загрузчиком операционной системы перед началом этапа инициализации ядра;
  - 1 — данная служба будет запускаться при инициализации ядра (подсистемой ввода/вывода);
  - 2 — служба будет запускаться диспетчером сервисов (smss.exe) при входе пользователя в систему;
  - 3 — данная служба запускается вручную в тот момент, когда она понадобится какой-нибудь программе;
  - 4 — служба не будет запускаться никогда.
- Type — этот параметр DWORD-типа указывает на то, к какому типу относится служба, и может принимать следующие значения:
  - 1 — служба определяет устройства уровня ядра;
  - 2 — служба определяет драйвер файловой системы;
  - 4 — служба является аргументом для адаптера;

- `8` — служба относится к службам файловой системы;
- `10` — служба является программой, запускающей свой процесс;
- `20` — служба является программой, запускающей общий процесс;
- `100` — если данная битовая маска присутствует в параметре `Type`, то система будет разрешать соответствующей службе взаимодействие с Рабочим столом (иначе служба не сможет вывести диалоговое окно, окно сообщения или свое окно).

Для примера попробуем зарегистрировать в системе свою собственную службу. Для этого достаточно только создать свой раздел в ветви `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services`, а в этом разделе создать такие параметры, как `ImagePath`, `DisplayName`, `Description`, `Group`. Результат можно видеть на рис. 8.1.

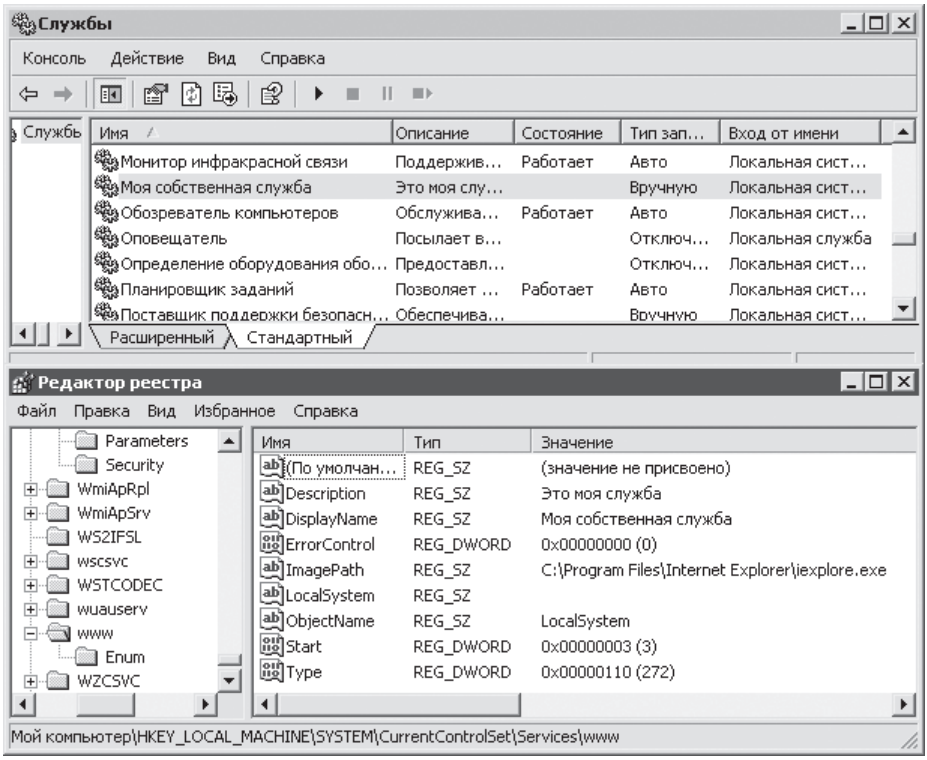


Рис. 8.1. Создание своей службы

Потенциально опасные ветви и параметры реестра

Данным разделом заканчивается знакомство с реестром Windows XP и параметрами, которые в нем могут находиться, поэтому сейчас хотелось бы перечислить

некоторые из ветвей реестра и параметров, которые если еще не используются, то скоро могут быть использованы вирусами, троянскими конями или просто различными программами-шутками для своей работы. В этом разделе будут также перечислены некоторые ветви реестра, создание или удаление которых может вызвать проблемы в работе операционной системы.

- HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\MiniNT — раздел не предназначен для операционной системы Windows XP, поэтому если он будет присутствовать в системе, то при каждой загрузке система будет выводить сообщение о нехватке размера файла подкачки pagefile.sys и создавать новый файл.
- HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\Desktop\NameSpace\{e17d4fc0-5564-11d1-83f2-00a0c90dc849} — об этом разделе уже упоминалось — если он окажется удаленным, то диалоговое окно Поиск работать не будет.
- HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\Desktop\NameSpace\{1f4de370-d627-11d1-ba4f-00a0c91eedba} — это еще один раздел, без которого не будет работать диалоговое окно Поиск.
- HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon — эта ветвь реестра может включать в себя множество параметров, за содержимым которых необходимо следить. Например, к ним можно отнести следующие параметры строкового типа.
  - System — определяет программы, которые будут запускаться с правами системы процессом WINLOGON.EXE при инициализации. Программы пишутся через запятую, то есть параметр может содержать вызов сразу нескольких программ. По умолчанию он ничему не равен.
  - Userinit — указывает программы, которые будут запускаться с правами пользователя при его регистрации процессом WINLOGON.EXE. Программы пишутся через запятую, это опять-таки означает, что в данной ветви могут находиться сразу несколько вызовов программ. По умолчанию значение данного параметра равно %systemroot%\system32\userinit.exe.
  - VmApplet — определяет программы, которые будут запускаться для настройки параметров виртуальной памяти процессом WINLOGON.EXE. Программы пишутся через запятую. По умолчанию значение данного параметра равно rundll32 shell32, Control_RunDLL "sysdm.cpl".
  - Shell — указывает файлы оболочки, которые будут запускаться при входе пользователя. Он как раз и определяет, что вы используете стандартную оболочку Windows explorer.exe — именно эта строка является значением параметра Shell по умолчанию. Но если вы измените значение этого параметра, например, на explorer.exe, notepad.exe, то наряду с оболочкой Windows при вашем входе в систему будет запускаться и Блокнот. Этот параметр может находиться как в корневом разделе HKEY_CURRENT_USER, так и в разделе HKEY_LOCAL_MACHINE.

  - GinaDLL — определяет путь к библиотеке msgina.dll, которая запускается вместе с системой по умолчанию и необходима для взаимодействия с оболочкой Windows. Если изменить значение этого параметра на вызов какой-нибудь программы, а не библиотеки, то при инициализации процесса WINLOGON.EXE будет выдано сообщение об ошибке и вы не сможете войти в систему.

- HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\ CurrentVersion\Windows — может содержать несколько потенциально опасных параметров, среди которых можно выделить следующие параметры строкового типа.
  - Run — определяет программы, которые будут запускаться с правами пользователя при его входе. Как и рассмотренные выше параметры, он может вызывать сразу несколько программ — в этом случае они пишутся через запятую. Параметр может находиться как в корневом разделе реестра HKEY_CURRENT_USER, так и в корневом разделе HKEY_LOCAL_MACHINE.
  - Load — указывает программы, которые будут запускаться с правами системы при входе любого пользователя. Как и рассмотренные выше параметры, он может вызывать сразу несколько программ — в этом случае они пишутся через запятую.
  - AppInit_DLLs — определяет библиотеки, необходимые для совместимости с каким-нибудь оборудованием или программой. Все описанные в данном параметре библиотеки будут запускаться перед запуском любой программы.
- HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\Browser Helper Objects — определяет все CLSID-номера ActiveX-объектов (в виде разделов, названных в честь CLSID-номера ActiveX-объекта), которые будут запускаться при каждом запуске браузера Internet Explorer.
- HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Session Manager — содержит REG_MULTI_SZ-параметр BootExecute, его значением являются команды, которые будут запускаться при каждой перезагрузке компьютера. Он используется системой для запуска таких системных программ работы с дисками, как автопроверка диска (значение этого параметра autocheck autochk *) или преобразование файловой системы диска FAT в NTFS (значение данного параметра autoconv \DosDevice\x: /FS:NTFS).
- HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Image File Execution Options — используется для возможности определения программ, при выполнении которых происходит утечка памяти. Но можно воспользоваться этой ветвью и для других целей. Например, если создать в ней раздел explorer.exe, а в нем создать DWORD-параметр ShutdownFlags и присвоить ему значение 3, то после выгрузки оболочки Windows существует вероятность, хотя и малая, что вы не сможете ее загрузить. Система может не дать вам этого сделать. Но даже если вы и сможете загрузить оболочку, то, скорее всего, увеличится количество ошибок неправильной адресации к памяти, выдаваемых различными программами.

- HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\SafeBoot\option — определяет, в каком режиме будет загружаться операционная система — обычном или безопасном. Именно поэтому возможна такая шутка — создайте в этой ветви реестра DWORD-параметр OptionValue и присвойте ему значение, равное 1. Теперь вы всегда будете загружаться в режиме, в чем-то подобном безопасному, — будет загружаться лишь минимальный набор сервисов, но драйверы устройств, таких как видеокарта, будут использоваться обычные, устанавливаемые вместе с устройством (а не стандартные, как при полноценном безопасном режиме). При этом, даже если вы являетесь администратором компьютера, вам будет запрещено запускать такие службы, как, например, Windows Audio, которые нельзя запускать в безопасном режиме. Раздел option создается только в безопасном режиме.
- HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\MountPoints\«çíà÷îê äèñê» и HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\MountPoints2\«çíà÷îê äèñê» — хранят настройки контекстного меню, значков дисков, а также описание файла autorun.inf, применявшегося ранее для запуска содержимого компакт-диска. В практике автора книги был такой случай, когда записи данных ветвей реестра постоянно приводили к отказу в доступе к приводу DVD. Другими словами, при попытке открытия содержимого диска, установленного в приводе DVD, отображался отказ в доступе к диску. При этом проблема решалась именно удалением раздела, названного в честь буквы диска, доступ к которому был отклонен (решалась до следующей попытки доступа к приводу). Поэтому, если у вас возникли подобные проблемы, просто попробуйте удалить соответствующие ветви реестра, а потом установить для них только доступ на чтение.

Часть 3

Консоль управления Microsoft

Глава 9

Окно консоли управления Microsoft

- **Запуск программы mmc.exe**
- **Окно программы mmc.exe**
- **Хранение параметров настройки консоли**
- **Добавление оснасток в консоль**

Консоль управления Microsoft — это специальное средство для администрирования компьютера, которое, начиная с Windows NT 4.0, пришло на смену Панели управления. Если раньше Панель управления содержала значительную часть программ, направленных на администрирование компьютера, то теперь эти программы преобразованы в так называемые оснастки, доступ к которым можно получить именно с помощью консоли управления Microsoft. Например, теперь именно с помощью оснасток можно выполнить такие операции, как добавление или удаление пользователя, дефрагментация диска, добавление или удаление общедоступных ресурсов, остановка или запуск служб и многое другое. При этом на основе наиболее часто используемых оснасток были созданы стандартные консоли (файлы с расширением MSC). Консоль — это набор оснасток, которые будут открываться при открытии консоли (при этом консоль будет открываться в программе `mmc.exe`, без которой работать с консолями невозможно). Другими словами, теперь с помощью консоли управления Microsoft можно с легкостью создавать собственные консоли, содержащие наиболее часто используемые вами оснастки.

Чтобы отобразить консоль управления Microsoft, необходимо в диалоговом окне Запуск программы ввести команду `mmc.exe`. Ввод данной команды приведет к отображению окна консоли управления Microsoft. Но еще несколько слов скажем о запуске программы `mmc.exe` — какие же процессы происходят на уровне файловой системы и реестра Windows XP при запуске консоли управления Microsoft?

Запуск программы mmc.exe

На уровне файловой системы, как оказывается, ничего интересного не происходит — по умолчанию консоль управления Microsoft не ведет журнал и не записывает события ни в один из журналов системы. Единственное, что можно отметить, так это запуск библиотеки `MMCNDMGR.DLL`, которая, как известно из части 1, является основной библиотекой консоли управления Microsoft и с помощью которой можно удалить или установить сведения о консоли управления в реестре. Например, с помощью команды `rundll32.exe MMCNDMGR.DLL, DllRegisterServer` выполняется повторная установка (в реестре) всех сведений, необходимых для запуска самой консоли управления Microsoft (в первую очередь сведений ветви системного реестра `HKEY_CLASSES_ROOT\CLSID\{43136EB5-D36C-11CF-ADBC-00AA00A80033}`, без которой работа с консолью управления Microsoft невозможна), а также выполняется повторная установка следующих стандартных оснасток: Элемент ActiveX, Ссылка на веб-ресурс и Папка.

На уровне реестра сначала идет обращение к ветви реестра `HKEY_CURRENT_USER\Software\Microsoft\Êîíñîëü óïðàâëåíèÿ (MMC)\Recent File List`, содержащей список всех ранее открывавшихся консолей. Эта ветвь включает в себя четыре параметра строкового типа с именами от `File1` до `File4`. Значения этих параметров определяют пути к уже открываемым вами консолям. Следует заметить, что программным способом не все открывавшиеся консоли записываются в эти четыре параметра, тем не менее на уровне реестра вы можете определить пути к любым консолям, с которыми часто работаете. Потом, например, можно запретить полный доступ к данной ветви реестра, оставив только доступ на чте-

ние, чтобы ссылки на часто открываемые вами консоли не заменялись другими ссылками.

Идет также обращение к ветви реестра Windows HKEY_CURRENT_USER\Software\Microsoft\Êîíñîëü óïðàâëåíèÿ (MMC)\Settings. Она содержит параметр строкового типа List Save Location, определяющий путь к каталогу, в котором по умолчанию будут сохраняться экспортируемые из консоли управления Microsoft данные.

После того как будет определен список ранее запускавшихся консолей, консоль управления Microsoft начнет считывать из реестра данные, необходимые для ее работы. Как правило, все эти данные расположены в корневом разделе системного реестра HKEY_CLASSES_ROOT. После определения конфигурации консоль управления Microsoft просматривает ветвь HKEY_CURRENT_USER\Software\Policies\Microsoft\MMC, которая может содержать ограничения групповой политики на запуск программы mmc.exe. Например, в этой ветви могут находиться следующие два параметра DWORD-типа (параметр появляется, если вы пользовались консолью управления Microsoft).

- RestrictAuthorMode — если значение равно 1, то запуск консоли управления Microsoft будет запрещен. При этом стоит заметить, что этот запрет не распространяется на консоли, созданные с помощью программы mmc.exe, — их по-прежнему можно будет открывать, но при этом работа в расширенном режиме (об этом ниже) будет невозможна.
- RestrictToPermittedSnapins — если значение этого параметра равно 1, то будет запрещен запуск всех консолей (точнее, всех оснасток), созданных с помощью консоли управления Microsoft. Саму же консоль управления Microsoft можно будет открывать, хотя, какой от нее толк, если открытие всех оснасток будет запрещено, неизвестно.

Консоль управления Microsoft также ищет DWORD-параметр Restrict_Run в ветвях реестра формата HKEY_CURRENT_USER\Software\Policies\Microsoft\MMC\{GUID-îñíàñòêè}. При этом если параметр Restrict_Run в одной из ветвей будет равен 1, то соответствующую оснастку запускать будет запрещено. Для примера работы данного ограничения можно создать параметр Restrict_Run в ветви реестра HKEY_CURRENT_USER\Software\Policies\Microsoft\MMC\{C96401CC-0E17-11D3-885B-00C04F72C717} и присвоить ему значение 1. После этого будет запрещено запускать оснастку Папки. Как правило, она всегда запускается вместе с созданными консолями.

Окно программы mmc.exe

Если приведенные выше параметры не существуют или равны нулю (точнее, параметр RestrictAuthorMode), то консоль управления Microsoft откроется и отобразит свое окно (рис. 9.1). Стоит заметить, что при открытии консоли управления Microsoft просто создается новая консоль с именем Консоль 1, что и можно увидеть на рис. 9.1, если посмотреть на строку заголовка окна.

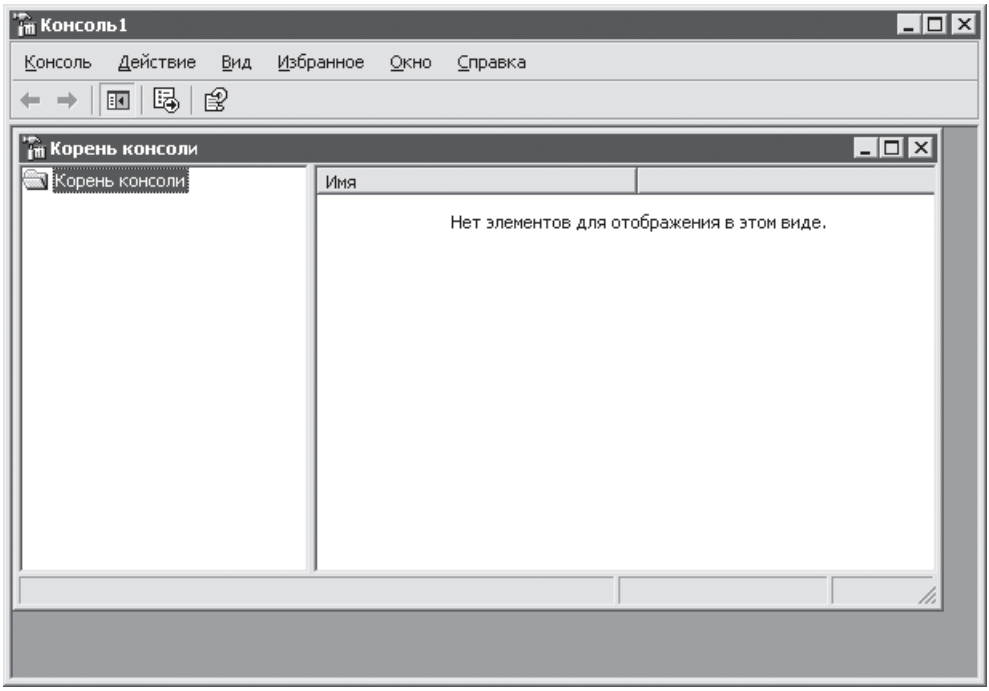


Рис. 9.1. Окно консоли управления Microsoft

Окно консоли управления Microsoft включает в себя вложенное окно, имеющее заголовок Корень консоли, с которым по умолчанию и будет выполняться работа (добавление или удаление оснасток, а также работа с содержимым оснастки). Но вы можете открыть еще одно окно Корень консоли. Для этого достаточно в меню Окно консоли управления Microsoft выбрать команду Новое (или нажать комбинацию клавиш Ctrl+W). Возможность создания отдельных окон в консоли была реализована для удобства работы с оснастками (чтобы не перегружать одно окно большим количеством загруженных оснасток). Например, в одно окно может быть загружена одна оснастка, в другое окно — несколько других оснасток и т. д., а переход между окнами можно выполнить с помощью меню консоли управления Microsoft Окно или с помощью выделения определенного окна мышью.

Существует также возможность определения отображаемых элементов создаваемой консоли. Для этого предназначена команда Настроить меню Вид. После вызова данной команды откроется диалог, который можно увидеть на рис. 9.2. С его помощью можно отобразить или скрыть определенные элементы окна консоли, просто сняв или установив флажок напротив их описания. При этом работа с данным диалогом не вызовет трудностей, так как при снятии или установке флажка в консоли автоматически скрывается или отображается соответствующий данному флажку элемент.

Но это еще не все команды для настройки вида создаваемой вами консоли. Например, в меню Действие можно выбрать команду Новый вид панели задач (панель

задач находится слева в окне, и по умолчанию на ней расположена только одна папка — Корень консоли). После ее выбора откроется Мастер создания вида панели задач (рис. 9.3), с помощью которого можно определить расположение панели задач в окне, а также варианты отображения элементов на ней.

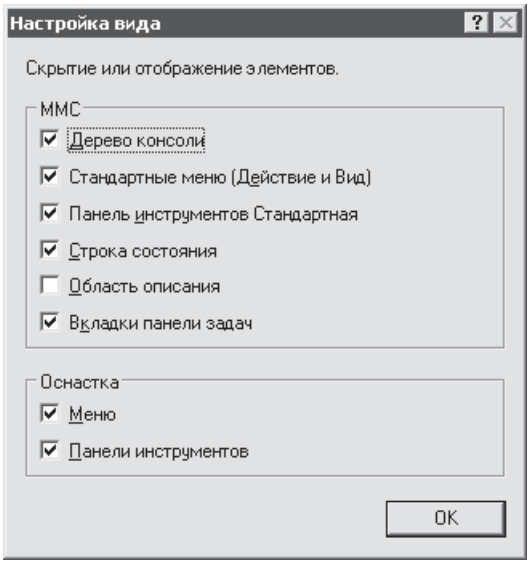


Рис. 9.2. Настройка отображения консоли

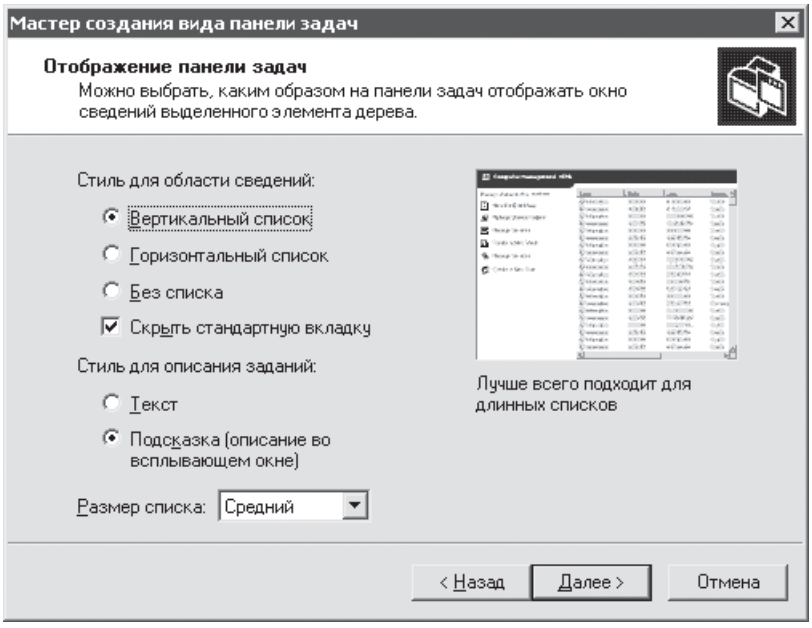


Рис. 9.3. Мастер настройки вида панели задач

С помощью меню Вид можно также определить режим вывода структуры оснасток (команды Крупные, Мелкие, Список, Таблица), но, поскольку вы еще не загрузили в созданную консоль ни одной оснастки, эти команды пока рассматривать не будем.

Если вы уже изменили некоторые настройки отображения элементов консоли с помощью диалога Параметры меню Вид, то изменения можно сохранить в меню Избранное. Например, так вы можете определить в данном меню несколько вариантов отображения консоли и при необходимости переходить между ними.

Хранение параметров настройки консоли

Стоит еще сказать о хранении настроек консоли. Если вы думаете, что настройки консоли хранятся в реестре, то это не так. На самом деле все настройки консолей содержатся в самих файлах консолей. Это легко понять на примере консоли, открытой в редакторе, подобном Блокноту. Это довольно важная особенность работы консоли, понимание которой очень важно для правильного ее использования.

Рассмотрим простой пример. Допустим, есть консоль, содержимое которой было скрыто с помощью диалога Параметры (рис. 9.4). Другими словами, все флажки, которые находятся в этом диалоге, были сняты.

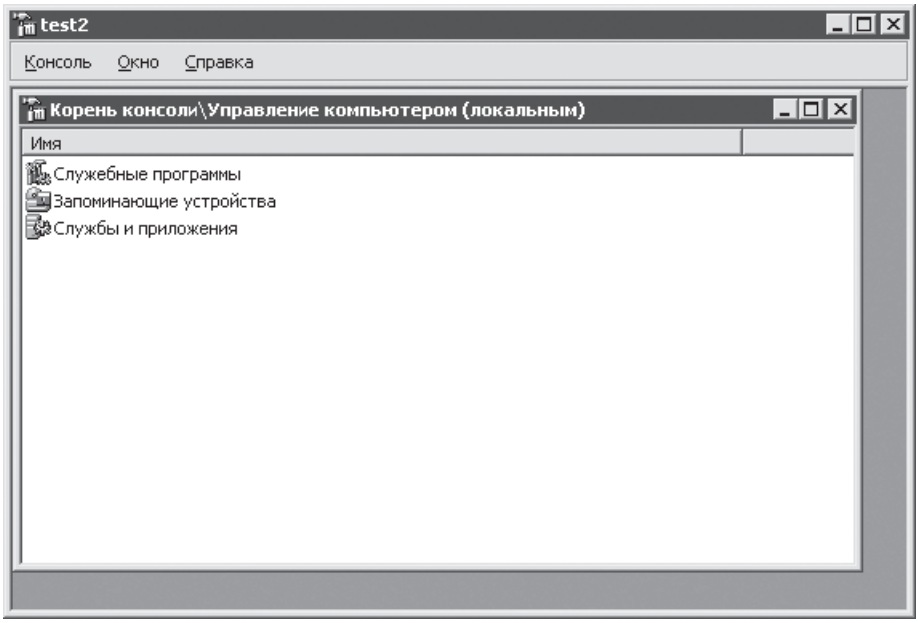


Рис. 9.4. Использование диалога Параметры для скрытия возможности настройки вида консоли

Если вы используете такой метод скрытия возможности изменения вида консоли, то посмотрим на содержимое данного файла консоли, открытого в Блокноте (рис. 9.5).

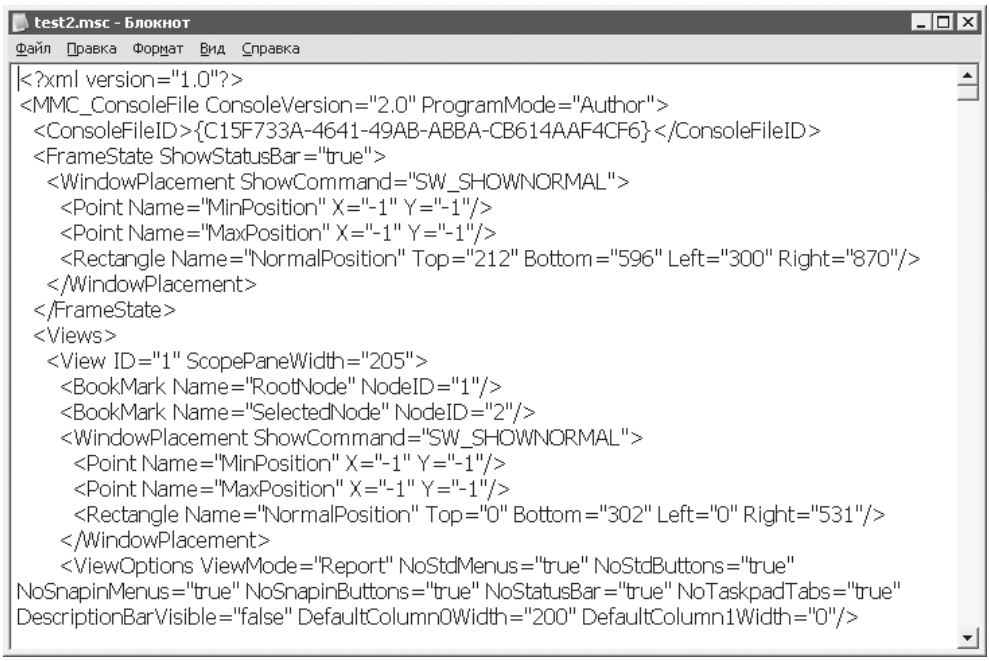


Рис. 9.5. Форматирование консоли

Как можно заметить, файлы консоли являются обычными файлами XML. А теперь посмотрите на нижнюю строку на рис. 9.5 — в этой строке определяется содержимое тега `ViewOptions`. Данный тег имеет следующие важные параметры:

- `NoStdMenus` — если значение равно `true`, то в меню консоли будут скрыты меню Действие, Вид и Избранное;
- `NoStdButtons` — если значение параметра равно `true`, то в консоли будет скрыта панель инструментов;
- `NoSnapinMenus` — если значение равно `true`, то меню оснасток будут скрыты (если, конечно, они имеют меню);
- `NoSnapinButtons` — если значение параметра равно `true`, то панели инструментов оснасток будут скрыты;
- `NoStatusBar` — если значение равно `true`, то строка состояния консоли будет скрыта;
- `NoTaskpadTabs` — если значение данного параметра равно `true`, то вкладки панели задач консоли будут скрыты;
- `DescriptionBarVisible` — если значение равно `false`, то область описания консоли будет скрыта.

Например, если присвоить параметру `NoStdMenus` значение `false`, то при следующем открытии данной оснастки меню Действие, Вид и Избранное опять отобразятся в строке меню.

Рассмотрим другой пример. В этом примере для ограничения возможностей работы консоли воспользуемся не только диалогом Параметры из меню Вид, но и диалогом Параметры, открыть который можно с помощью команды Параметры меню Консоль (рис. 9.6). С помощью данного диалога можно изменить значок консоли, а также определить режим ее отображения (список Режим консоли): Авторский, Пользовательский — полный доступ, Пользовательский — ограниченный доступ, много окон, Пользовательский — ограниченный доступ, одно окно. Режим Авторский используется по умолчанию и позволяет выполнять любые команды, доступные с помощью консоли. Режим Пользовательский — полный доступ позволяет выполнить любые доступные в консоли возможности, но запрещает добавление в консоль новых оснасток. Режим Пользовательский — ограниченный доступ, много окон запрещает добавление в консоль новых оснасток, а также закрытие окон консоли (при этом новые окна будет разрешено создавать). Режим Пользовательский — ограниченный доступ, одно окно запрещает добавление в консоль новых оснасток, а также использование в консоли более одного окна.

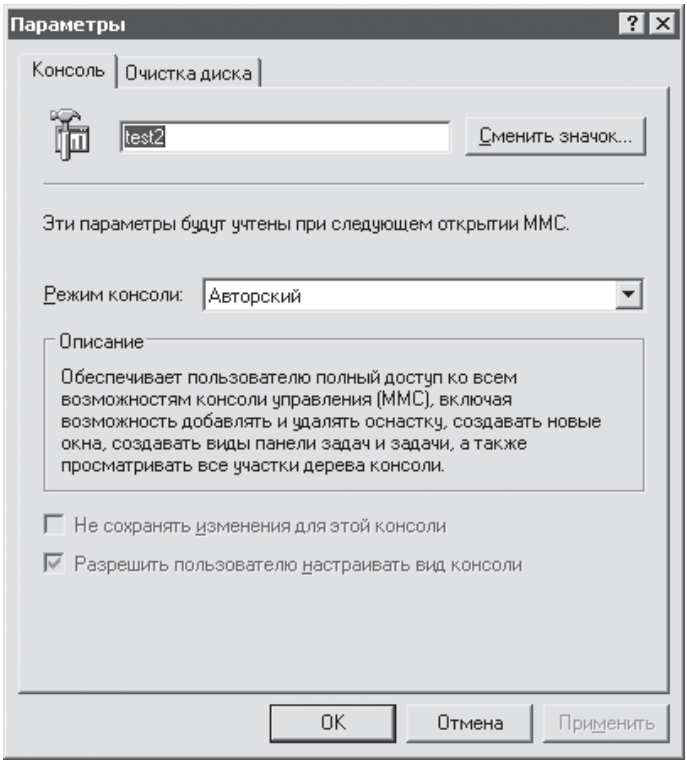


Рис. 9.6. Диалог настройки ограничений консоли

Если будет выбран один из пользовательских режимов отображения консоли, то в диалоге Параметры также станут доступны флажки Не сохранять изменения для этой консоли и Разрешить пользователю настраивать вид консоли. Из названия флажков понятно, для ограничения чего они используются.

На рис. 9.7 можно увидеть результат использования режима Пользовательский — ограниченный доступ, одно окно. Как можно заметить, возможность добавления и удаления оснасток, а также возможность создания новых окон были скрыты. Тем не менее команда Параметры из меню Консоль осталась, но при ее открытии окажется, что она не будет содержать вкладки Консоль.

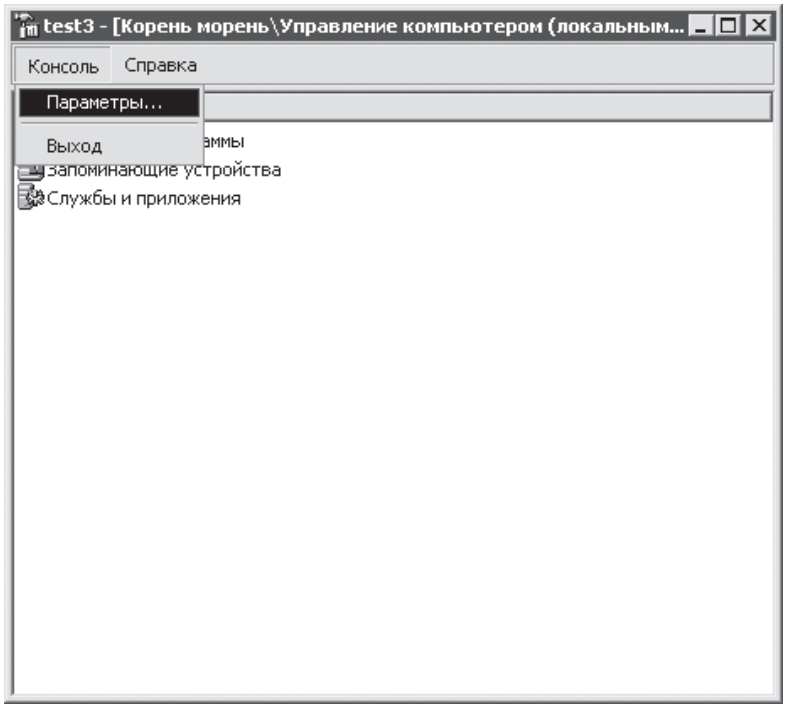


Рис. 9.7. Применение пользовательского режима

Итак, что же изменилось при использовании режима Пользовательский — ограниченный доступ, одно окно в содержимом файла консоли? На рис. 9.8 можно увидеть изменения, которые произошли в верхнем теге `MMC_ConsoleFile`. Одним из его параметров является `ProgramMode`, который в данном случае равен `UserSDI`. Следует заметить, что при авторском режиме доступа к оснастке этот параметр равен `Author`. Другими словами, если вы измените значение параметра `ProgramMode` на `Author`, то при следующем запуске консоли попадете в авторский режим с возможностью добавления новых оснасток.

Как можно заметить, параметры ограничений консолей обходятся довольно просто. Поэтому если вы будете создавать ограниченные оснастки для пользователей (как советуют многие администраторы), то не забудьте установить с помощью ACL (вкладка Безопасность диалога свойств оснастки) только доступ на чтение и исполнение консоли для пользователя, которому создаете оснастку (если, конечно, консоль не помещается в папку, от которой она будет наследовать такие права).

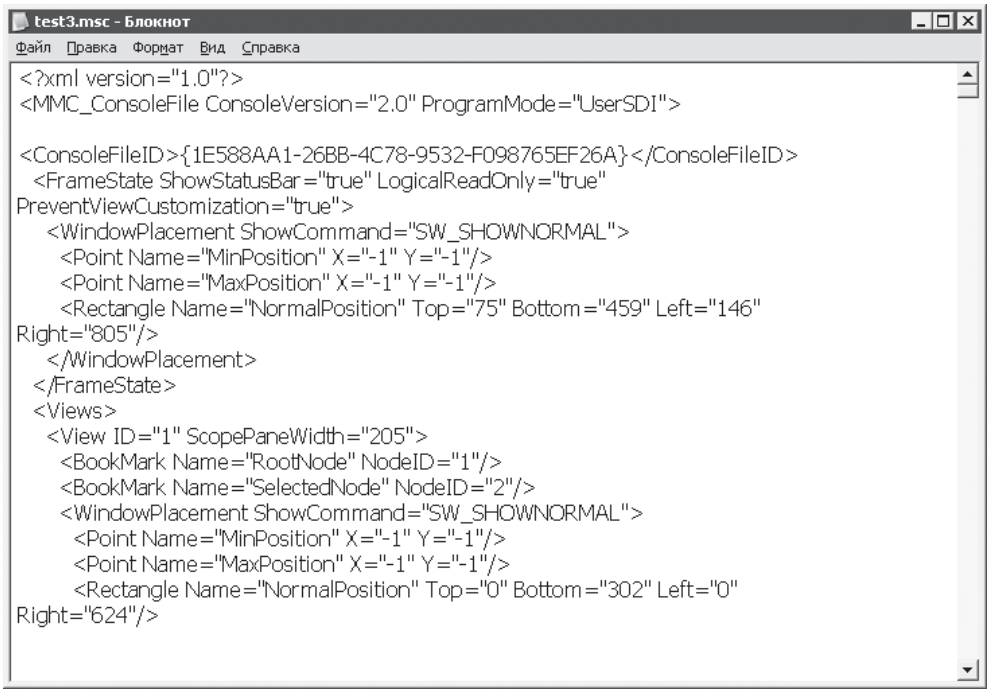


Рис. 9.8. Применение пользовательского режима консоли

Добавление оснасток в консоль

Теперь попробуем загрузить какую-нибудь оснастку в созданную консоль. Для этого необходимо воспользоваться командой Добавить или удалить оснастку из меню Консоль (или комбинацией клавиш Ctrl+M). После вызова этой команды перед вами отобразится диалоговое окно Добавить/удалить оснастку (рис. 9.9), с помощью которого можно добавить в консоль новую оснастку или удалить уже присутствующие. Чтобы добавить консоль, нужно нажать кнопку Добавить.

После нажатия кнопки Добавить консоль управления Microsoft начнет просматривать содержимое ветви реестра `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\MMC\SnapIns`. Она хранит ссылки на все GUID-номера оснасток, доступных на компьютере. Именно список оснасток из этой ветви и будет отображаться в появившемся после нажатия кнопки Добавить диалоговом окне. Иными словами, если удалить из данной ветви GUID-номер оснастки, то ее нельзя будет открыть с помощью данного списка, хотя оснастка по-прежнему будет работать в уже готовых консолях. Но для удаления оснастки из списка можно поступить проще — дело в том, что все оснастки, которые будут отображаться в списке Добавить изолированную оснастку, должны содержать в своей ветви вложенный раздел `StandAlone`. Если его удалить, то оснастка не будет отображаться в списке Добавить изолированную оснастку. Можно же, наоборот, добавить этот раздел к одной из ветвей, в которой его не существует. Например, если добавить его к разделу `{243E20B0-48ED-`

11D2-97DA-00A024D77700}, то появится возможность включать в консоль оснастку Модуль расширения съемных носителей. А если добавить его к разделу {BACF5C8A-A3C7-11D1-A760-00C04FB9603F}, то появится возможность добавления к консоли оснастки Установка программ (пользователи). Можно также добавить оснастку Установка программ (Компьютеры). Для этого необходимо воспользоваться разделом {942A8E4F-A261-11D1-A760-00C04FB9603F}. Две предыдущие возможности понадобятся в следующих главах книги, ведь по умолчанию никаким другим способом нельзя получить доступ к оснасткам Установка программ (пользователи) и Установка программ (Компьютеры), если компьютер не находится в домене.

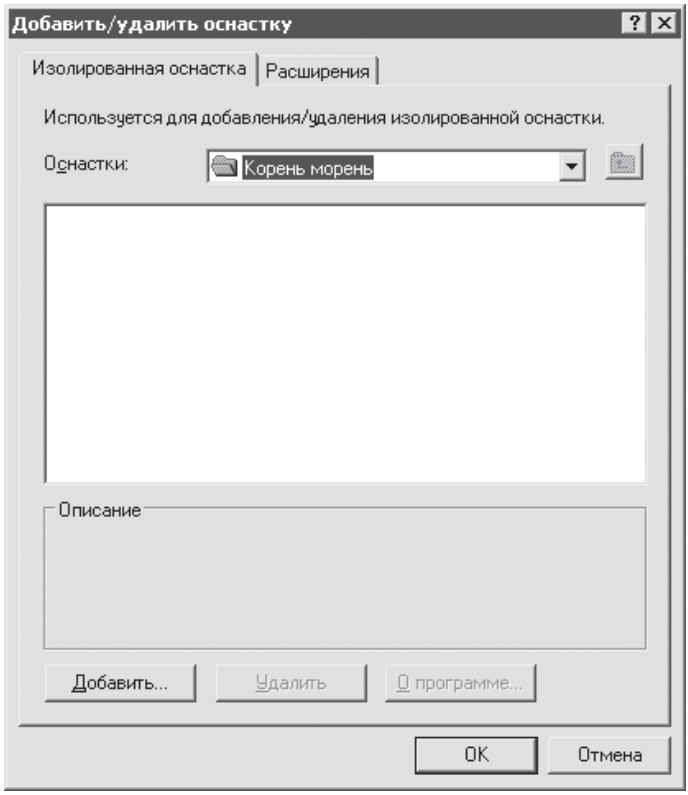


Рис. 9.9. Диалог добавления оснасток в консоль

С помощью данной ветви можно также изменить название оснастки, отображаемое в списке оснасток, — оно хранится в параметре, имеющем строковый тип NameStringIndirect ветви HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\MMC\SnapIns\{GUID-íîìåð îñíàñòêè}. Например, чтобы изменить название оснастки Диспетчер устройств на Описание установленного на компьютере оборудования, необходимо присвоить новое название оснастки параметру NameStringIndirect ветви реестра HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\MMC\SnapIns\{74246bfc-4c96-11d0-abef-0020af6b0b7a} (рис. 9.10).

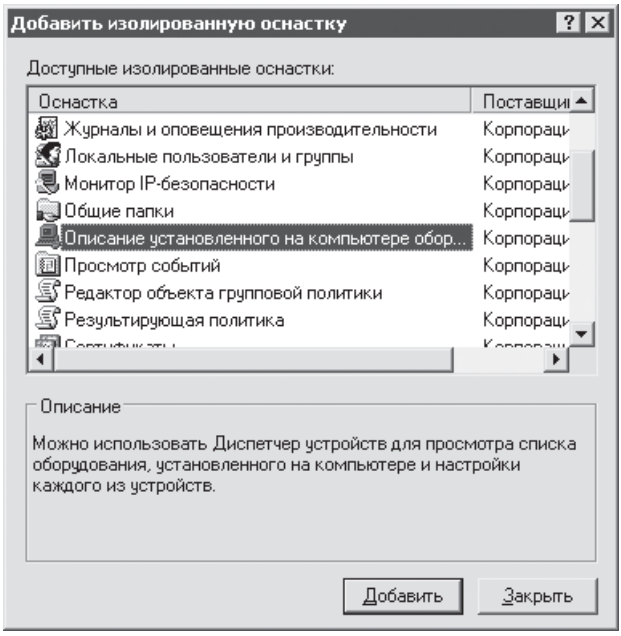


Рис. 9.10. Изменение названия оснастки Диспетчер задач

ПРИМЕЧАНИЕ

Существует еще один трюк, который можно выполнить с помощью ветви реестра HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\MMC\SnapIns\{GUID-номер оснастки}. Это запрет отображения диалога Добавить/удалить оснастку. Для этого достаточно изменить название одной из ветвей реестра {GUID-номер оснастки} на название {{GUID-номер оснастки}. Например, если изменить название уже рассмотренного раздела {74246bfc-4c96-11d0-abef-0020af6b0b7a} на {{74246bfc-4c96-11d0-abef-0020af6b0b7a}, то при выборе команды Добавить или удалить оснастку из меню Консоль ничего происходить не будет.

Для примера попробуем добавить в консоль оснастку Редактор объекта групповой политики. Для этого необходимо выделить в списке эту оснастку, нажать кнопку Добавить, после чего появится окно Выбор объекта групповой политики, в котором нужно нажать кнопку Готово. Далее нужно нажать кнопку Закрыть в окне Добавить изолированную оснастку, чтобы закрыть список оснасток. В диалоге Добавить/удалить оснастку появилась добавленная оснастка. В этом окне нужно нажать кнопку ОК. Теперь предлагаю посмотреть на рис. 9.11. Слева на этом рисунке отображена консоль в расширенном виде. Этот вид стал доступен в Windows XP и отличается от обычного тем, что слева появилась панель описания элемента оснастки. Очень часто эта панель мешает просмотру оснастки, поэтому приходится переходить к обычному виду (по умолчанию оснастка загружается в расширенном виде). Справа же на рисунке отображена та же оснастка, но в обычном виде — при этом возможность расширенного вида была удалена. Чтобы удалить возможность рас-

ширенного вида, необходимо удалить из ветви реестра `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\MMC\SnapIns` раздел `{B708457E-DB61-4C55-A92F-0D4B5E9B1224}`.

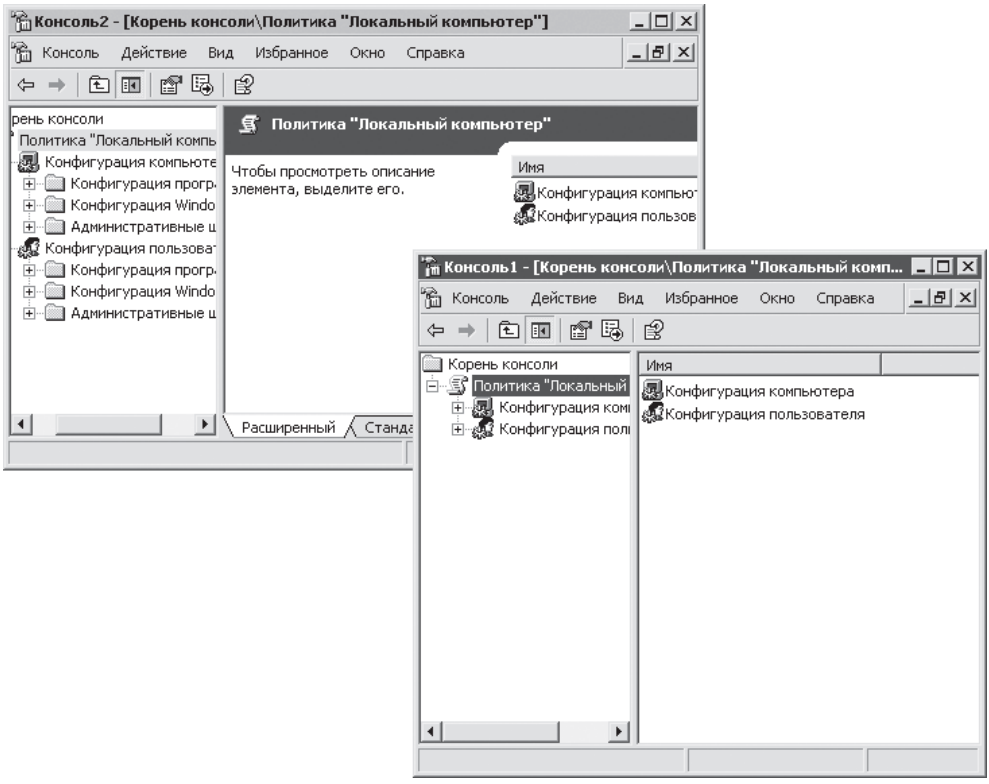


Рис. 9.11. Расширенный и обычный вид консоли

Напоследок рассмотрим краткое описание всех оснасток, доступных в операционной системе Windows XP. Далее они будут подробно описаны.

- Дефрагментация диска — позволяет узнать, необходима ли диску дефрагментация, а также выполнить ее. Для открытия данной оснастки можно воспользоваться стандартной консолью Windows XP `dfrg.msc`.
- Диспетчер устройств — дает возможность просмотреть конфигурацию оборудования, установленного на компьютере, а также удалить, обновить или откатить драйверы для конкретного компонента компьютера. Оснастку можно вызвать с помощью консоли `devmgmt.msc`.
- Журналы и оповещения производительности — с помощью данной оснастки можно задать ведение журнала производительности или трассировки, а также указать механизм оповещения администратора при возникновении определенного события. Оснастку можно вызвать в виде стандартной консоли Производительность — `perfmon.msc`.

- Управляющий элемент WMI — позволяет настроить параметры работы элемента WMI, который добавляет новые функциональные возможности серверу сценариев Windows. Для открытия данной оснастки можно воспользоваться консолью Инструментарий управления WMI, вызов которой осуществляется с помощью команды `wmimgmt.msc`.
- Управление компьютером — содержит набор оснасток, с помощью которых можно выполнить большинство административных работ на локальном компьютере: Просмотр событий, Общие папки, Диспетчер устройств и многие другие. Оснастку можно вызвать как одноименную консоль `compmgmt.msc`.
- Локальные пользователи и группы — позволяет добавить или удалить группы пользователей, доступные на компьютере, а также добавить, удалить или отредактировать свойства учетной записи пользователей, зарегистрированных на компьютере. Для открытия оснастки можно воспользоваться одноименной консолью, вызов которой осуществляется с помощью команды `lusrmgr.msc`.
- Общие папки — с помощью данной оснастки можно просмотреть все общедоступные папки системы, а также добавить или удалить общедоступную папку. Можно также настроить параметры доступа к общедоступным папкам или просмотреть открытые в данный момент общедоступные ресурсы и сеансы подключения. Для открытия оснастки можно воспользоваться одноименной консолью, вызов которой осуществляется с помощью команды `fsmgmt.msc`.
- Папка — является стандартной оснасткой, устанавливаемой вместе с консолью управления Microsoft и позволяющей упорядочить содержимое консоли в случае, когда она включает в себя большое количество оснасток.
- Просмотр событий — с помощью этой оснастки можно просмотреть стандартные журналы системы. Оснастку можно вызвать в виде стандартной консоли `eventvwr.msc`.
- Редактор объекта групповой политики — позволяет настроить параметры пользователя или компьютера, направленные на ограничения предоставляемых функций. Оснастку можно вызвать с помощью консоли `gpedit.msc`.
- Результирующая политика — дает возможность просмотреть установленные для компьютера или пользователя параметры ограничений групповой политики. Оснастку можно вызвать в виде стандартной консоли `rsop.msc`.
- Сертификаты — с помощью данной оснастки можно просмотреть доступные на компьютере сертификаты и их свойства, а также выполнить поиск среди сертификатов. Для ее открытия можно воспользоваться одноименной консолью, вызов которой осуществляется с помощью команды `certmgr.msc`.
- Служба индексирования — позволяет настроить службу индексирования или найти с ее помощью файлы по части их содержимого. Для открытия данной оснастки можно воспользоваться консолью `ciadv.msc`.
- Службы — с помощью данной оснастки можно отключить, запустить, приостановить или продолжить работу большинства служб, установленных на компьютере, а также настроить их запуск. Для открытия оснастки можно воспользоваться консолью `services.msc`.

- Ссылка на веб-ресурс — с помощью этой оснастки можно загрузить в консоль локальную HTML-страницу или веб-сайт для его просмотра (как в браузере). Она является стандартной оснасткой, устанавливаемой вместе с консолью управления Microsoft.
- Управление политикой безопасности IP — позволяет настроить правила передачи сообщений по протоколу IP, а также различные фильтры пакетов для этого протокола. Оснастка является частью консоли Групповая политика, вызвать которую можно с помощью команды `gpedit.msc`.
- Управление съемными носителями — с помощью этой оснастки можно просмотреть список подключенных к компьютеру съемных носителей, а также извлечь эти носители. Для открытия оснастки можно воспользоваться консолью Съемные носители, вызов которой осуществляется с помощью команды `ntmsmgr.msc`.
- Управление дисками — позволяет отформатировать диск, сделать его активным, изменить букву локального диска, а также открыть его содержимое в Проводнике. Для ее открытия можно воспользоваться консолью Управление дисками, вызов которой осуществляется с помощью команды `diskmgmt.msc`.
- Элемент ActiveX — является стандартной оснасткой, устанавливаемой вместе с консолью управления Microsoft.
- Шаблоны безопасности — с помощью данной оснастки можно создать собственный шаблон безопасности или воспользоваться стандартными шаблонами для быстрой настройки параметров безопасности компьютера.

Глава 10

Оснастки настройки Windows XP

- **Дефрагментация диска**
- **Диспетчер устройств**
- **Служба индексирования**
- **Службы**

Дефрагментация диска

Как уже было сказано, оснастка Дефрагментация диска предназначена для выполнения дефрагментации и входит в состав консоли dfrg.msc.

Оснастка имеет GUID-номер {43668E21-2636-11D1-A1CE-0080C88593A5}, то есть если вы создадите DWORD-параметр Restrict_Run в ветви реестра HKEY_CURRENT_USER\ Software\Policies\Microsoft\MMC\{43668E21-2636-11D1-A1CE-0080C88593A5} и присвоите ему значение 1, то будет запрещено открывать оснастку Дефрагментация диска.

ПРИМЕЧАНИЕ

Оснастка может не работать и в случае повреждения ветви системного реестра HKEY_CLASSES_ROOT\AppID\{80EE4901-33A8-11d1-A213-0080C88593A5}. Например, если данная ветвь реестра будет содержать параметр строкового типа RunAs с некорректным значением, то такие функции оснастки, как анализ и дефрагментация дисков, работать не будут.

Запуск оснастки

При открытии оснастки система начинает просматривать необходимые для работы библиотеки, название одной из которых определено в параметре строкового типа ResourceDllName, расположенным в ветви реестра HKEY_LOCAL_MACHINE\ SOFTWARE\Microsoft\Dfrg\ResourceDllName.

По умолчанию значение данного параметра равно %systemroot%\system32\ DfrgRes.dll и определяет название библиотеки, хранящей ресурсные записи оснастки Дефрагментация диска (различный текст, используемый для построения этой оснастки). В данной ветви реестра могут также содержаться следующие параметры.

- CreateLogFile — этот параметр DWORD-типа определяет, будет ли создаваться файл журнала анализа диска. Если его значение равно 1, то файл журнала создаваться будет.
- LogFilePath — параметр строкового типа, определяет название файла журнала (и путь к нему), в который будет записываться информация о выполнении анализа диска, если значение параметра CreateLogFile будет равно 1.

Если работа оснастки не запрещена групповыми политиками и все библиотеки и параметры реестра, необходимые оснастке, содержат корректные данные, то после ввода в диалоге Запуск программы команды dfrg.msc (или открытия этой оснастки с помощью консоли управления Microsoft mmc.exe) перед вами отобразится окно, подобное приведенному на рис. 10.1.

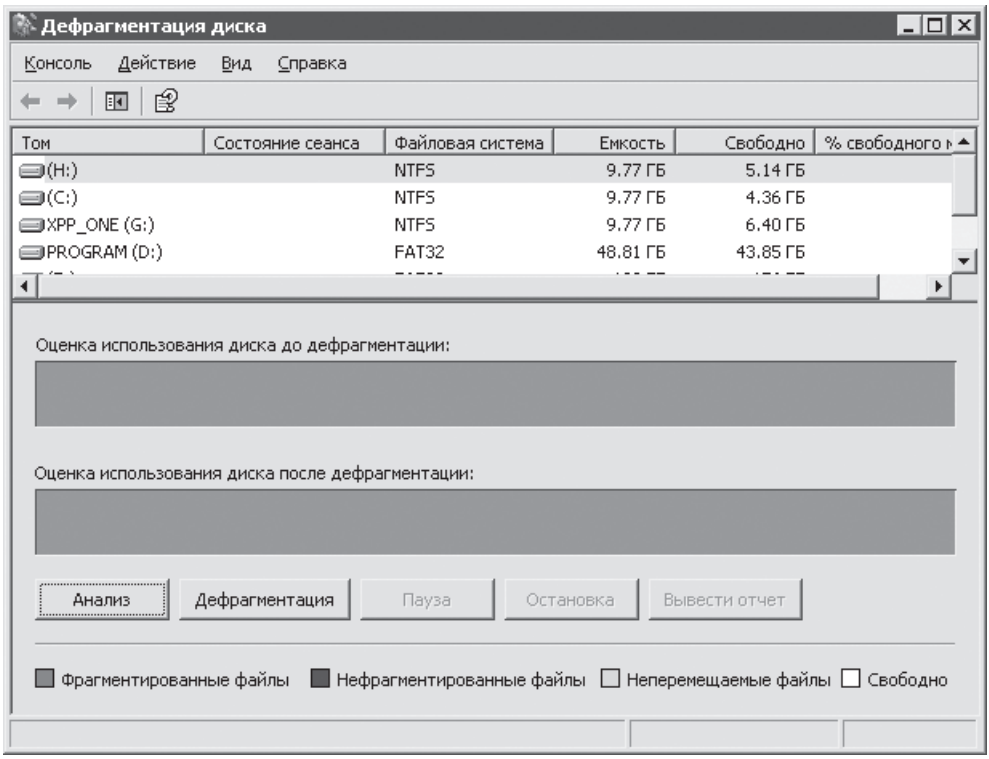


Рис. 10.1. Окно оснастки Дефрагментация диска

Работа с оснасткой

Окно оснастки Дефрагментация диска состоит из двух областей. Верхняя область содержит список всех логических дисков, установленных на компьютере (как жестких, так и съемных дисков). В этой области можно узнать букву диска, файловую систему, используемую на нем, его полный объем, а также свободное место на диске (как в гигабайтах, так и в процентном соотношении). В нижней области расположены кнопки для работы с оснасткой, а также две полосы для оценки использования диска — одна определяет примерное расположение содержимого на диске до выполнения дефрагментации (активизируется после нажатия кнопки Анализ), а вторая определяет оценку расположения содержимого диска после дефрагментации (активизируется после проведения дефрагментации).

Кнопки в нижней области определяют полную функциональность оснастки Дефрагментация диска. Иными словами, с ее помощью можно только выполнить анализ расположения содержимого диска и дефрагментацию диска. При этом обе эти возможности по умолчанию используют для своей реализации программу `DfrgNtfs.exe`. Эта программа запускается как ActiveX-объект, и сведения о ней находятся в ветви реестра `HKEY_CLASSES_ROOT\CLSID\{80EE4901-33A8-11d1-A213-0080C88593A5}\LocalServer32`. Параметр `(Ïî óìîë÷àíèþ)`

этой ветви как раз и содержит название программы — `DfrgNtfs.exe`. Данная ветвь системного реестра может также включать в себя параметр строкового типа `ServerExecutable`. Он определяет название программы, запускающейся как сервер для программы `DfrgNtfs.exe`. Иными словами, если значение параметра `ServerExecutable` будет равно, допустим, `cmd.exe`, то после нажатия кнопки Анализ или кнопки Дефрагментация будет запущена программа `cmd.exe` (при этом в заголовке командной строки будет указано название `DfrgNtfs.exe`).

После выделения в верхней области необходимого логического диска и нажатия кнопки Анализ консоль управления Microsoft начнет проверку расположения файлов на данном диске. После проверки перед вами будет отображено сообщение о том, нужна ли данному логическому диску дефрагментация или нет. При этом окно будет содержать три кнопки: Вывести отчет (выводит полную информацию о логическом диске (размер кластера, количество фрагментированных файлов фрагментация MFT и т. д.), а также список наиболее сильно фрагментируемых файлов, если вы используете возможность ведения файла журнала с помощью описанного выше параметра `CreateLogFile`, то та же информация будет храниться в файле журнала), Дефрагментация (выполнить дефрагментацию диска) и Закрыть. После проведения анализа полоса оценки расположения файлов будет включать в себя гистограмму содержимого логического диска (рис. 10.2).

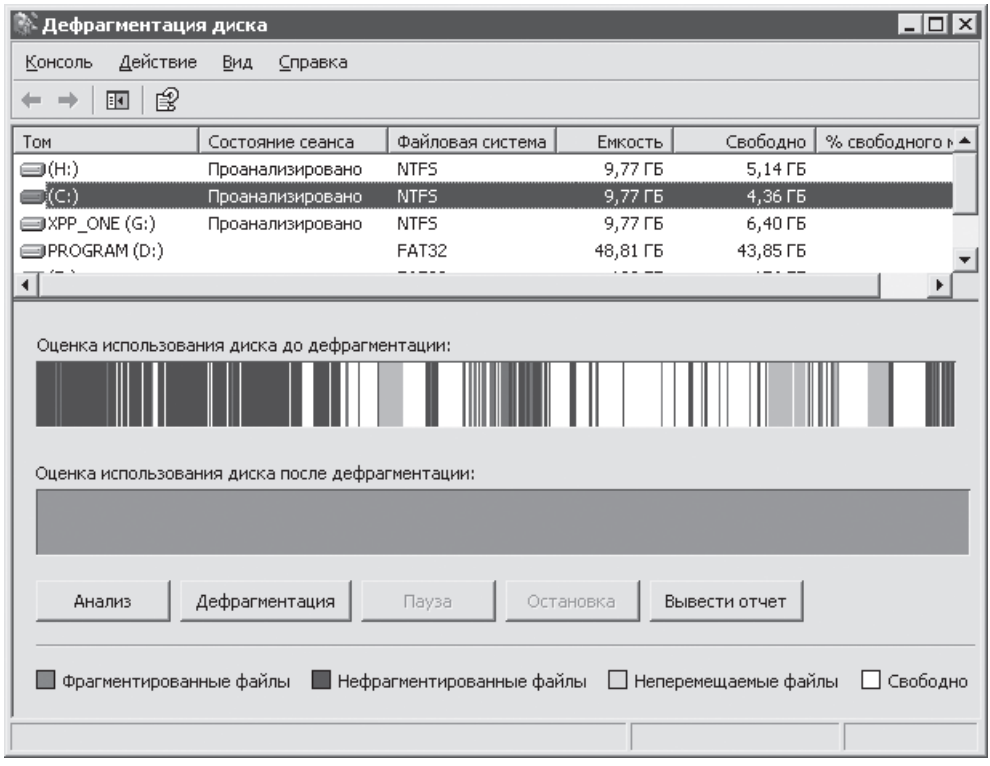


Рис. 10.2. Окно оснастки после проведения анализа логического диска

ПРИМЕЧАНИЕ

MFT — главная файловая таблица, содержащая сведения обо всех файлах, расположенных на логическом диске. По умолчанию MFT занимает для своего роста 12 % от всего объема логического диска.

ПРИМЕЧАНИЕ

Запрещено проводить анализ или дефрагментацию логического диска, если системой для него была определена необходимость проверки с помощью команды chkdsk /f.

Если после анализа вы решили провести дефрагментацию диска, необходимо нажать кнопку Дефрагментация. Но перед этим рекомендуется выполнить следующие действия.

1. Удалить все ненужные вам файлы, расположенные на данном диске.
2. Удалить все временные файлы и папки (обычно папки для temp-файлов имеют название temp, но перед удалением папки стоит просмотреть ее содержимое и уже на его основе решить, удалять папку или нет).
3. Удалить файлы журнала, хранящиеся на данном логическом диске.
4. Проанализировать расположение содержимого диска с помощью кнопки Анализ.

Диспетчер устройств

Диспетчер устройств входит в стандартную консоль `devmgmt.msc` и имеет GUID-номер `{74246BFC-4C96-11D0-ABEF-0020AF6B0B7A}`. После вызова данной консоли откроется окно, подобное приведенному на рис. 10.3.

Окно Диспетчера устройств отображает все установленное на компьютере оборудование. Но иногда бывают ситуации, когда оборудование, подключающееся в «горячем» режиме (то есть без выключения компьютера), не будет распознано Диспетчером устройств. При этом оно будет считаться неустановленным и работать не будет. Например, довольно часто этим грешат модемы, подключаемые в процессе работы компьютера (особенно часто могут не определяться подключаемые в «горячем» режиме мобильные телефоны, используемые как модемы для подключения к Интернету с помощью стандарта GPRS, хотя обычные модемы также иногда не распознаются). Как правило, это не такая страшная проблема — скорее всего, нераспознанное оборудование будет найдено при обновлении списка оборудования Диспетчера устройств. Для инициации процесса обновления необходимо в меню Действие выбрать команду Обновить конфигурацию оборудования. После этого консоль управления Microsoft начнет поиск новых устройств Plug and Play.

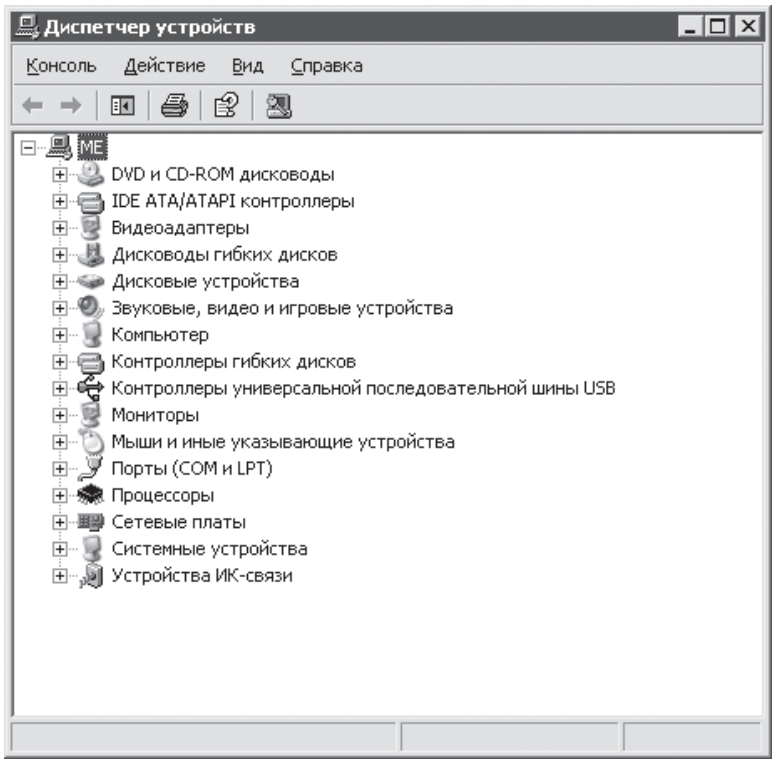


Рис. 10.3. Окно консоли devmgmt.msc

По умолчанию оборудование группируется по типу устройств, но можно определить другой способ группировки оборудования. Для этого применяется меню Вид консоли Диспетчер устройств. С помощью данного меню можно использовать следующую группировку:

- Устройства по типу — используется по умолчанию и группирует устройства по их типу (например, все сетевые карты (как физические, так и виртуальные) в группе Сетевые платы);
- Устройства по подключению — группирует все устройства по интерфейсу подключения, используемого ими (например, все устройства, подключенные к шине PCI);
- Ресурсы по типу — группирует все устройства по типу ресурсов, которые они используют (то есть если устройство использует как адреса памяти, так и прерывание IRQ, то оно будет описано сразу в двух группах);
- Ресурсы по подключению — ресурсы, как и в предыдущем способе, группируются в четыре группы (Ввод/вывод, Запрос на прерывание (IRQ), Память и Прямой доступ к памяти), но теперь ресурсы в группах дополнительно группируются по диапазону адресов (прерываний и т. п.), который они используют.

В меню Вид присутствует флажок Показать скрытые устройства, установка которого приводит к отображению в окне консоли Диспетчер устройств списка системных устройств и драйверов, а также устройств, отключенных или не работающих в данный момент. Отдельно стоит сказать о типе устройств Драйверы устройств не Plug and Play, отображаемом при установке флажка Показать скрытые устройства. Данный тип содержит список всех драйверов устройств не Plug and Play, установленных на компьютере. Причем диалог Свойства для устройств данного типа является единственным способом отключения таких драйверов — для этого используется раскрывающийся список Тип (в области Автозагрузка) на вкладке Драйвер. На этой вкладке можно также определить раздел ветви реестра HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services, в котором хранятся настройки данного драйвера.

ПРИМЕЧАНИЕ

Следует внимательно относиться к драйверам типа устройств Драйверы устройств не Plug and Play, так как довольно часты случаи запуска вирусов, троянских коней, перехватчиков клавиатуры и других «хакерских» программ, выдающих себя за драйверы устройств не Plug and Play.

Пример диалога Свойства

Для примера рассмотрим диалог Свойства какого-нибудь устройства. Для этого будет использоваться стандартный способ группировки устройств. Например, выберите группу DVD и CD-ROM дисководы. Если вы имеете несколько дисководов такого типа, то данная группа будет содержать несколько устройств. Чтобы просмотреть свойства устройства, нужно дважды щелкнуть кнопкой мыши на нем, после чего перед вами отобразится диалог свойств, подобный приведенному на рис. 10.4.

Вкладка Общие, как правило, стандартна для всех устройств, установленных на компьютере. Данная вкладка описывает следующие данные.

- Тип устройства — в этой строке описывается класс, к которому принадлежат устройства, указанные в данной группе. Как правило, тип устройства является названием группы, в которой оно описано в оснастке Диспетчер устройств. При этом название типа устройства хранится в реестре — для этого применяется параметр (Ïî óìîë÷àíèþ) ветвей реестра формата HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class\{GUID-íîìåð êëàññà óñòðîéñòâà}. Например, рассматриваемый тип устройств DVD и CD-ROM дисководы описывается в параметре (Ïî óìîë÷àíèþ) ветви системного реестра HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class\{4D36E965-E325-11CE-BFC1-08002BE10318}. Если вы измените значение параметра (Ïî óìîë÷àíèþ) данной ветви реестра, то соответственно изменится и название типа устройств в Диспетчере устройств.

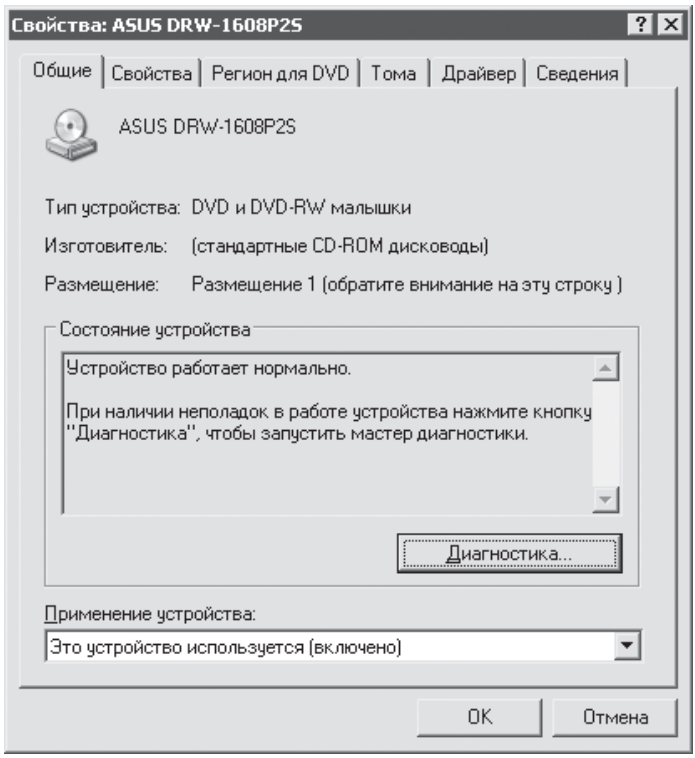


Рис. 10.4. Отображение диалога свойств устройства

ПРИМЕЧАНИЕ

В ветви реестра HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\ Class\ {GUID-номер устройства} может также присутствовать DWORD-параметр NoDisplayClass. Именно он и определяет, будет ли считаться данное устройство скрытым. Другими словами, например, если параметр NoDisplayClass будет присутствовать в ветви реестра HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class\{4D36E965-E325-11CE-BFC1-08002BE10318}, то тип DVD и CD-ROM дисководы по умолчанию будет скрыт в Диспетчере устройств и увидеть его можно будет, только установив флажок Показать скрытые устройства в меню Вид.

В данной ветви реестра может также содержаться DWORD-параметр NoUseClass. Если он будет присутствовать в ветви описания класса устройств, то сведения об устройствах данного типа вообще будут скрыты из консоли Диспетчер устройств. Например, чтобы скрыть описываемую группу DVD и CD-ROM дисководы, нужно создать параметр NoUseClass в ветви реестра HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\ Class\{4D36E965-E325-11CE-BFC1-08002BE10318}.

- Изготовитель — определяет производителя данного устройства. Как правило, если производителем является Microsoft (или производитель вообще не описывается),

то пишется, что данное устройство является стандартным. Название производителя определяется в параметре строкового типа `ProviderName` ветви реестра формата `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class\{GUID-íîìåð êëàññà óñòðîéñòâà}\«íîìåð óñòðîéñòâà»`.

- Размещение — указывает шину или порядковый номер, по которому установлено устройство. Сведения о размещении считываются в процессе загрузки компьютера, но существует возможность добавления к этим сведениям своей строки (или замещения сведений своей строкой). Для этого применяется параметр строкового типа `LocationInformationOverride` ветви реестра формата `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class\{GUID-íîìåð êëàññà óñòðîéñòâà}\«íîìåð óñòðîéñòâà»`. Например, сведения в скобках в строке Размещение на рис. 10.4 были добавлены (для некоторых устройств они замещают оригинальные сведения, а для некоторых добавляются к оригинальным сведениям) с помощью строкового параметра `LocationInformationOverride` ветви реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class\{4D36E965-E325-11CE-BFC1-08002BE10318}\0000`.
- Состояние устройства — содержит информацию о неполадках в работе устройства или, если неполадок нет, строку Устройство работает нормально.
- Применение устройства — позволяет отключить или включить устройство.

К другим возможностям, которые можно настроить на этой вкладке, является стандартный значок для данной группы устройств (отображается напротив названия дисковода). Его идентификатор определен в параметре строкового типа `Icon` ветви реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class\{4D36E965-E325-11CE-BFC1-08002BE10318}` (только идентификатор, а не название библиотеки и идентификатор). Например, по умолчанию для устройств типа DVD и CD-ROM дисководы используется значок с идентификатором `-51`.

Вкладка Свойства хранит сведения о настройке различных функций устройства. Например, для рассматриваемого дисковода могут содержаться параметры настройки уровня громкости при воспроизведении музыки с помощью дисковода, а также флажок, позволяющий использовать цифровое воспроизведение музыки с данного дисковода.

Вкладка Регион для DVD определяет текущий регион (регионы можно менять всего пять раз), предназначенные для которого DVD вы можете просматривать. Содержимое данной вкладки используется специальными лицензионными DVD, которые записывались для определенного региона страны. При этом следует серьезно отнестись к смене регионов, ведь, как утверждается на данной вкладке, больше пяти раз регион изменить будет нельзя (даже если вы переустановите операционную систему).

Вкладка Драйвер содержит сведения о драйвере, установленном для данного устройства, имя поставщика драйвера (параметр строкового типа ProviderName), дату разработки драйвера (REG_BINARY-параметр DriverDateData), версию драйвера (параметр строкового типа DriverVersion). Все эти параметры хранятся в ветвях формата HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class\{GUID-íîìåð êëàññà óñòðîéñòâà}\«íîìåð óñòðîéñòâà». На вкладке Драйвер можно выполнить обновление или удаление текущего драйвера, а также его откат. Откат применяется в случае, если после обновления драйвера устройство стало работать некорректно, и позволяет установить тот драйвер, который применялся до обновления.

Вкладка Сведения позволяет просмотреть служебную информацию об устройстве. По умолчанию отображается Код экземпляра устройства, но с помощью раскрывающегося списка данной вкладки можно просмотреть очень многие характеристики устройства. Большая часть из них предназначены не для пользователя (то есть описывается специальными константами, которые могут знать разве что представители технического персонала производителя устройства или очень опытные пользователи), но некоторые могут быть интересны и нам. Например, элемент Служба данного списка определяет название раздела в ветви системного реестра HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services, который определяет настройки службы, реализующей функции данного устройства. Элементы списка, названия которых начинаются со слова Зависимости, определяют оборудование (службы и т. п.), которое должно быть удалено перед тем, как выполнить ту или иную операцию с устройством. Элементы Установщик классов и Соустановщики классов списка определяют функции библиотек (вспомните вызов команды rundll32.exe), предназначенные для установки GUID-номера класса (разделы ветви реестра HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class) или соустановщика класса.

Дополнительные настройки диалогов Свойства

С помощью системного реестра существует возможность изменения содержимого диалогового окна Свойства для некоторых устройств. Вкратце рассмотрим эти возможности.

Если на вашей материнской плате присутствуют USB-порты, то в консоли Диспетчер устройств будет присутствовать тип устройств Контроллеры универсальной последовательной шины USB. Среди устройств, подключенных к этому типу, будут присутствовать корневые USB-концентраторы. Если открыть диалог Свойства одного из корневых USB-концентраторов, то можно увидеть вкладку Управление электропитанием, с помощью которой настраивается возможность отключения устройства для экономии энергии. С помощью реестра существует возможность скрыть эту вкладку (или, наоборот, добавить ее, если она отсутствует). Для того чтобы скрыть вкладку Управление электропитанием, достаточно параметру DWORD-типа DisableSelectiveSuspend из ветви реестра HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\usb присвоить значение, равное 1.

Если на вашем компьютере присутствует сетевая карта, то в консоли Диспетчер устройств будет существовать тип устройств Сетевые платы. Диалог Свойства устройств данного типа содержит вкладку Дополнительно, с помощью которой можно настроить дополнительные свойства работы сетевой карты. При этом данная вкладка будет включать в себя поле свойств, а также список, в котором указываются значения для этих свойств. Список значений свойств хранится в одной из ветвей реестра, которые будут рассмотрены далее. Все эти ветви содержат параметры строкового типа 1, 2 и т. д. Значения этих параметров как раз и находятся в списках.

Рассмотрим одно из свойств настройки сетевой карты и ветвь реестра, в которой содержатся значения списка для него (данные свойства являются аппаратно-зависимыми, то есть в зависимости от возможностей вашей сетевой карты могут применяться различные свойства, поэтому нет смысла рассказывать обо всех, так как нет гарантий, что ваша сетевая карта будет использовать эти свойства).

- `802.1p QOS` — определяет, будет ли использоваться резервирование 20 % пропускной способности сети для сервиса QOS. По умолчанию значение данного свойства равно `Disable`. Как же это влияет на реестр? Во-первых, настройки отображения данного дополнительного параметра сетевой карты расположены в ветви реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class\{4D36E972-E325-11CE-BFC1-08002bE10318}\«íîìåðóñòðîéñòâà»\Ndi\params\Enable8021p`. Если удалить данную ветвь реестра, то исчезнет и сама возможность настройки сервиса QOS. Кроме того, ветвь содержит следующие параметры.
  - `type` — этот параметр строкового типа определяет способ отображения значения для данного свойства сетевой карты. По умолчанию значение данного параметра равно `enum`, что говорит об использовании списка для представления значений. Значение может быть равно `edit`. В этом случае для представления значений будет применяться поле, в котором пользователь сам должен ввести необходимое ему значение.
  - `ParamDesc` — данный параметр строкового типа определяет название свойства сетевой карты. Иными словами, для нашего свойства сетевой карты данный параметр реестра будет равен `802.1p QOS`. Вы можете изменить значение этого параметра на более понятное, например на значение `Âêëþ÷èòü ñåðâèñ QOS`.
  - `Default` — параметр строкового типа, определяет номер значения (например, если используется значение, описанное параметром `0`, то значение этого параметра будет равно `0`) свойства настройки сетевой карты, используемого в данный момент. При этом сами возможные значения хранятся в разделе `enum` данной ветви реестра. Именно этот раздел и содержит рассмотренный выше список параметров формата 1, 2 и т. д. По умолчанию в разделе `enum` находится только два параметра, имеющих значения `Enable` и `Disable`. Но вы можете создать новые параметры, например третий па-

раметр 3, которому присвоить значение, допустим, Âùå íå ðåøèë. После этого список значений для данного свойства настройки сетевой платы будет содержать и ваше значение (рис. 10.5). Если же для ввода значений свойства настройки сетевой карты используется поле ввода, то данный параметра будет хранить само введенное пользователем значение.

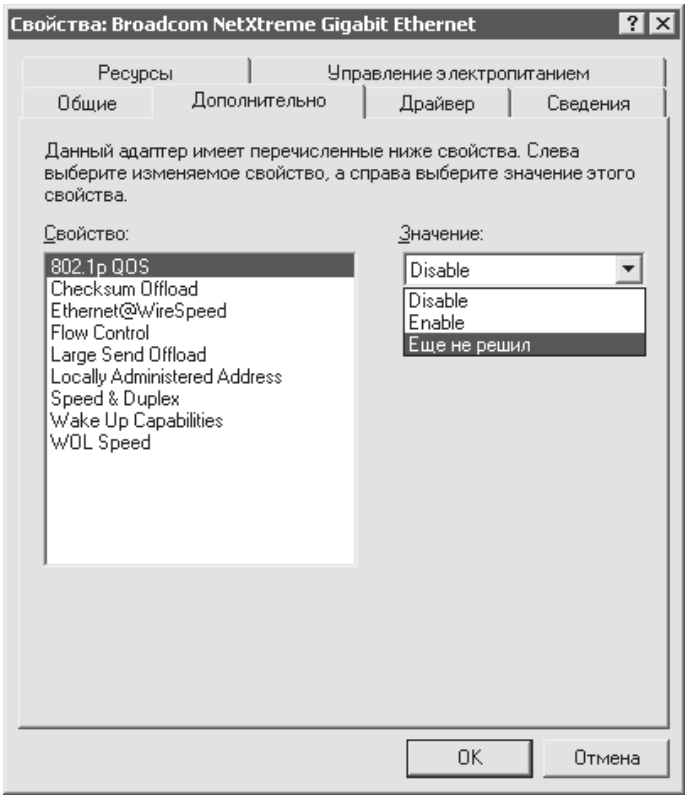


Рис. 10.5. Редактирование списка дополнительных параметров настройки сетевой карты

Изменение данного свойства влияет на DWORD-параметр Enable8021p, расположенный в ветви реестра HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class\{4D36E972-E325-11CE-BFC1-08002bE10318}\«íîìåð óñòðîéñòâà». Если это свойство отключено, то данный параметр будет равен 0.

ПРИМЕЧАНИЕ

Почему ноль? Именно потому, что значение свойства Disabled описано в параметре 0. Например, если выбрать созданное значение Еще не решил, которое описано в параметре 3, то параметру Enable8021p будет присвоено значение 3.

Если на вашем компьютере присутствует устройство инфракрасной связи (IrDA), то в консоли Диспетчер устройств будет находиться тип устройств Устройства ИК-связи. Диалог Свойства для устройств данного типа содержит вкладку Настройка инфракрасной связи, с помощью которой можно указать скорость передачи данных по инфракрасной связи. При этом возможные значения скорости передачи данных можно определить с помощью реестра. Для этого применяются два параметра REG_MULTI_SZ-типа из ветви системного реестра HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class\{6BDD1FC5-810F-11D0-BEC7-08002BE2092F}\«íîìåð óñòðîéñòâà». Параметр MaxConnectList содержит список возможных скоростей, который будет отображаться на вкладке Настройка инфракрасной связи. Значения данного параметра играют косметическую роль, то есть присутствуют только на вкладке, а не отражают реальную скорость передачи данных. Реальные же скорости передачи данных, которые соответствуют значениям предыдущего параметра, находятся в параметре MaxConnectRate.

Диалог Свойства для устройств ИК-связи может содержать вкладку Дополнительно. Свойства, отображаемые на этой вкладке, будут описываться в ветви системного реестра HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class\{6BDD1FC5-810F-11D0-BEC7-08002BE2092F}\«íîìåð óñòðîéñòâà»\Ndi\Params. Формат описания данных свойств был рассмотрен выше.

Другие настройки реестра, изменяемые оснасткой

Теперь вкратце рассмотрим настройки других диалогов Свойства, представляющих интерес с точки зрения их взаимодействия с реестром Windows XP.

- Чтобы запретить вывод сообщений об ошибках в работе устройств, подключенных к портам USB, необходимо DWORD-параметру ErrorCheckingEnabled присвоить значение 0. Этот параметр расположен в ветви системного реестра HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Usb (по умолчанию раздел не существует).
- Чтобы запретить отключение питания неиспользуемого USB-контроллера, необходимо присвоить DWORD-параметру HcDisableSelectiveSuspend значение, равное 1. Параметр может находиться в ветвях реестра, имеющих формат HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class\{36FC9E60-C465-11CF-8056-444553540000}\«íîìåð êîíòðîëëåðà». Чтобы определить, какой именно USB-контроллер описывается в данной ветви реестра, необходимо посмотреть на параметр строкового типа DriverDesc.
- Чтобы изменить текущую скорость порта для модема, нужно воспользоваться DWORD-параметром MaximumPortSpeed, расположенным в ветви реестра HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class\{4D36E96D-E325-11CE-BFC1-08002BE10318}\«íîìåð óñòðîéñòâà». Его значение определяет скорость работы порта для модема и может принимать следующие значения: 12c, 4b0, 960, 12c0, 2580, e100, 1ñ200, 35400, 70800 и т. д.

- Для того чтобы изменить название журнала (и путь к нему), предназначенного для протоколирования работы модема, необходимо воспользоваться параметром строкового типа `LoggingPath`, который расположен в ветви реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class\{4D36E96D-E325-11CE-BFC1-08002BE10318}\«íîìåð óñòðîéñòâà»`. Стоит также учитывать, что если `REG_BINARY`-параметр `Logging` из данной ветви реестра будет равен `0`, то протоколирование работы модема вестись не будет.
- Чтобы изменить дополнительные параметры инициализации модема, необходимо воспользоваться параметром строкового типа `UserInit`. Параметр расположен в ветви реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class\{4D36E96D-E325-11CE-BFC1-08002BE10318}\«íîìåð óñòðîéñòâà»` и содержит строку инициализации модема.
- Параметры работы COM-портов также можно изменить. Для этого предназначена ветвь реестра `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Ports`. Она включает в себя список параметров строкового типа, среди которых можно найти такие параметры, как, например, `COM1:`, `COM2:` и т. д. Эти параметры как раз и определяют настройки соответствующих COM-портов и содержат значения такого формата: `ñêîðîñòü, ÷åòíîñòü, áèòû äàííûõ, ñòîïîâûå áèòû, óïðàâëåíèå ïîòîêîì`. Четность может принимать следующие значения:
  - `n` — нет;
  - `e` — чет;
  - `o` — нечет;
  - `m` — маркер;
  - `s` — пробел.

  Управление потоком может содержать следующие значения:
  - `p` — аппаратное;
  - `x` — Xon/Xoff;
  - значение отсутствует — нет.

  Например, значение параметра `9600,n,8,1` расшифровывается так: скорость передачи данных равна 9600 бит/сек, четность не применяется, на представление одного символа используется 8 бит, интервал времени между передаваемыми символами равен 1 бит/сек.

Служба индексирования

Раньше служба индексирования уже рассматривалась с точки зрения параметров реестра, влияющих на ее производительность. Сейчас же будет рассмотрена оснастка Служба индексирования: как с ее помощью определить каталоги для индексирования,

а также выполнить поиск в содержимом индексированных файлов. Оснастка Служба индексирования входит в состав консоли `ciadv.msc` и имеет GUID-номер `{95AD72F0-44CE-11D0-AE29-00AA004B9986}`.

Но перед кратким описанием консоли `ciadv.msc` поговорим о том, что же можно ожидать от этой службы. Служба индексирования при нахождении нового файла просматривает список фильтров документов для определения, умеет ли она работать с файлами данного расширения. По умолчанию в состав операционной системы Windows XP входят фильтры для работы с документами, создаваемыми продуктами комплекта Microsoft Office, а также с текстовыми файлами, файлами HTML и файлами почты и групп новостей. При этом любая программа может поставлять собственные фильтры для работы с новыми расширениями файлов. Если найденный файл имеет расширение, поддерживаемое существующими в операционной системе фильтрами, то служба индексирования пытается определить язык, на котором написан данный файл. Русского языка служба индексирования не понимает, поэтому поиск по русским словам невозможен. Зато поиск в каталоге службы индексирования возможен по английским словам. Поэтому если содержимое файла написано на английском языке, то оно разбивается на отдельные слова и заносится во временный список слов. В процессе слияния списки слов помещаются в общий индекс на жестком диске в сильно сжатом виде. Благодаря этому возможно увеличение скорости поиска в содержимом файлов в несколько раз. Как можно заметить, единственным минусом службы индексирования является непонимание русского языка, поэтому если вы наиболее часто имеете дело с файлами на русском языке, то использовать службу индексирования нет смысла. Если же вы также имеете дело с файлами на английском языке, то рекомендуется помещать их в одну папку, а после этого с помощью консоли Служба индексирования указать системе выполнять индексацию только содержимого этой папки.

После запуска консоли `ciadv.msc` перед вами отобразится окно, включающее в себя один элемент — System (если на компьютере присутствует IIS-сервер, то в оснастке также будет присутствовать элемент Web). Элемент System представляет собой стандартный каталог, создаваемый службой индексирования при установке операционной системы. Именно с помощью элемента System и выполняется по умолчанию вся работа со службой индексирования на файловой системе Windows.

Контекстное меню элемента System содержит пункт Свойства, после выбора которого перед вами отобразится диалог для настройки некоторых параметров работы службы индексирования. Например, с помощью данного диалога можно настроить следующие параметры: определить, будут ли добавляться в индекс псевдонимы папок общего доступа, указать, будут ли индексироваться файлы с неизвестным расширением, определить, будут ли генерироваться аннотации к индексам. Некоторые из этих параметров уже рассматривались в виде параметров реестра.

С помощью контекстного меню элемента System можно остановить или запустить службу индексирования, а также выполнить слияние. Слиянием называется объединение нескольких списков слов и сохраненных ранее индексов (в буфере) в по-

стоянный индекс. Выполнение слияния может сильно загрузить процессор, поэтому выполнять его нужно в нерабочее время.

С помощью контекстного меню элемента System можно также создать новую папку (внутри элемента System) и определить, будет ли содержимое этой папки добавляться к индексу. Для этого служит команда Создать ▸ Папка. Но перед созданием папки щелкните дважды кнопкой мыши на элементе System. После этого отобразится содержимое данного элемента: разделы Папки, Свойства и Опрос каталога.

С помощью раздела Папки можно определить, содержимое каких каталогов будет индексироваться, а каких в индекс помещаться не будет. По умолчанию в индекс помещается содержимое всех логических дисков компьютера, кроме папок профилей пользователей Application Data и Local Settings (в индекс также не входит содержимое съемных дисков). Вы можете добавить свои папки или удалить уже существующие. С помощью контекстного меню папок можно также выполнить повторное сканирование их содержимого на предмет существования новых файлов, которые можно поместить в индекс (команда Все задачи ▸ Выполнить сканирование, которая появляется только тогда, когда Служба индексирования запущена).

С помощью раздела Свойства можно определить индексируемые свойства файлов, а также указать кэш, в котором будут храниться данные свойства. Существует два вида кэша свойств — первичный и вторичный. Первичный кэш свойств обеспечивает максимальную скорость поиска, но имеет небольшой размер, поэтому в него нужно помещать лишь часто используемые свойства (желательно помещать в него свойства постоянной длины, иначе работа с первичным кэшем может замедлиться). Вторичный же кэш используется для хранения остальных свойств. Раздел Свойства содержит вложенные разделы, идентифицирующие одно свойство. Контекстное меню этих разделов включает в себя команду Свойства, с помощью которой можно отобразить одноименный диалог Свойства. С помощью этого диалога можно определить кэш, в котором будет храниться данное свойство, указать размер, резервируемый в кэше для элементов данного свойства, а также определить тип свойства, если система сделала это неправильно (если свойство имеет тип VT_LPWSTR, это говорит о том, что система не знает истинного типа данного свойства).

С помощью раздела Опрос каталога можно перейти на форму для поиска в содержимом индексированных документов (рис. 10.6). Форма находится отдельно в каталоге %systemroot%\HELP и называется ciquery.htm (еще в приведенном каталоге хранится ciquery.htm, также используемый разделом Опрос каталога). Иными словами, если данные HTML-файлы будут отсутствовать в каталоге %systemroot%\HELP, то вы не сможете обратиться к содержимому каталога службы индексирования. С помощью данной формы можно выполнить поиск по ключевым словам (не забудьте, что русские слова не индексируются) или по свойствам. Более подробную информацию о поиске в каталоге можно найти в файле справки по данной консоли.

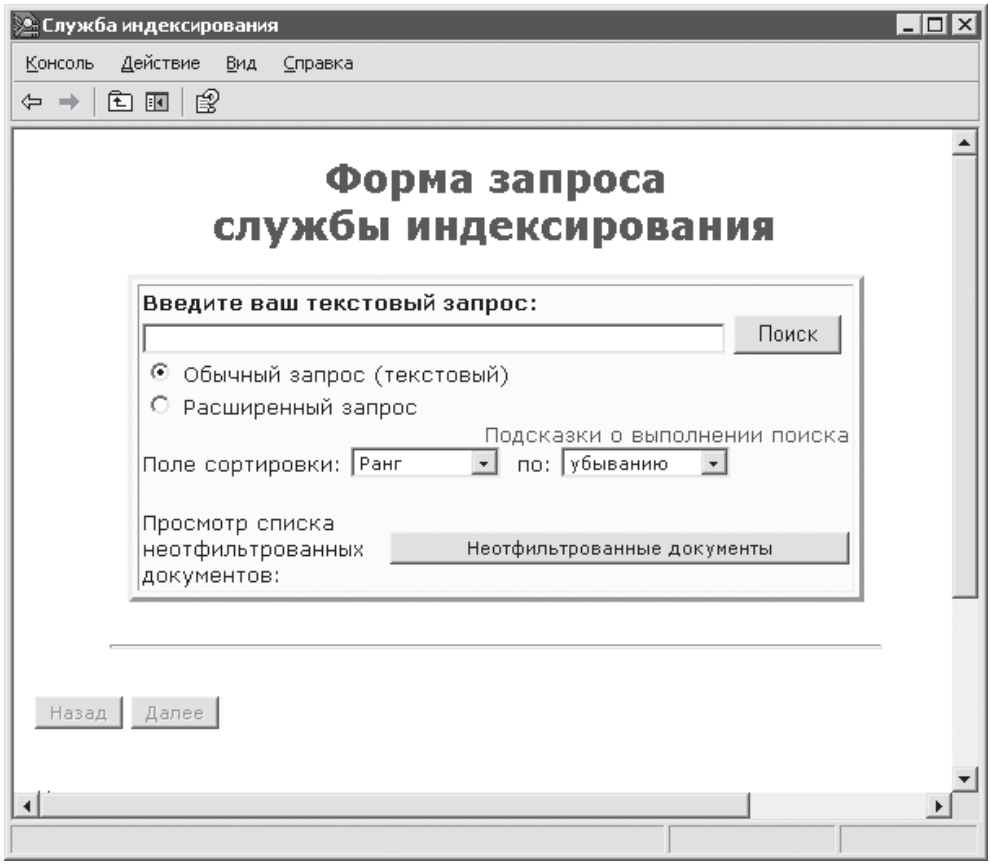


Рис. 10.6. Поиск в каталоге индексированных файлов

Службы

В части 2 уже рассматривались как способ описания служб в реестре, так и сами службы, запускающиеся автоматически при запуске компьютера (а также ветви реестра и иногда параметры реестра, используемые этими службами). Сейчас же рассмотрим более простой способ доступа к службам — с помощью оснастки Службы. Эта оснастка входит в стандартную консоль `services.msc` и имеет CLSID-номер `{58221C66-EA27-11CF-ADCF-00AA00A80033}`.

После запуска консоли перед вами отобразится окно со списком служб, доступных на данном компьютере. Это не полный список служб — в нем не указаны системные драйверы и другие службы, которые система считает драйверами. Тем не менее эта оснастка является основным способом работы со службами. С ее помощью также можно просмотреть список служб, запущенных на другом компьютере. Для этого предназначена команда Подключиться к другому компьютеру в меню Действия. С помощью меню Действия можно также отослать сообщение пользователю другого

компьютера сети. Для этого необходимо воспользоваться командой Все задачи ▸ Отправка сообщения консоли из меню Действия. После этого консоль управления Microsoft предложит вам ввести само сообщение, а также выбрать компьютер (или компьютеры), пользователю которого нужно отослать сообщение.

Но вернемся к списку служб. Он отображен на правой панели консоли, которая, в свою очередь, содержит следующие столбцы, описывающие службу.

- Имя — указывает имя службы и является значением параметра системного реестра `DisplayName` раздела службы.
- Описание — определяет описание того, для чего предназначена данная служба, и является значением параметра реестра `Description` раздела службы.
- Состояние — указывает, запущена ли в данный момент служба.
- Тип запуска — определяет, как запускается служба, и может принимать следующие значения: Отключено, Авто (запускается вместе с системой) и Вручную (запускается по требованию других программ или служб). Данный столбец для определения типа запуска использует параметр реестра `Start` раздела службы.
- Вход от имени — указывает учетную запись, от имени которой будет выполняться запуск службы. Службу можно запускать или от имени любого пользователя системы, или от имени трех стандартных учетных записей компьютера: Локальная система (с правами системы), Локальная служба (с правами группы Пользователи) и Сетевая служба (с правами группы Пользователи). Данный столбец для определения прав службы использует параметр реестра `ObjectName` раздела службы.

Кроме просмотра состояния службы, с помощью консоли Службы можно остановить, запустить или приостановить работу службы. Для этого предназначены, соответственно команды Пуск, Стоп и Продолжить контекстного меню конкретной службы (некоторые стандартные службы запрещено останавливать). В контекстном меню служб также присутствует команда Свойства. С ее помощью можно отобразить диалог для настройки параметров запуска службы, который содержит следующие вкладки.

- Общие — с помощью данной вкладки можно просмотреть название службы, ее описание, путь к файлу службы, а также можно задать тип запуска службы либо вообще отключить ее запуск.
- Вход в систему — позволяет определить учетную запись, от имени которой будет запускаться служба. При этом следует учитывать, что только служба, запущенная от имени системы, может взаимодействовать с Рабочим столом пользователя. Это не очень хорошо с точки зрения безопасности, так как желательно, чтобы как можно меньше служб запускалось с привилегиями системы, а если службе необходимо взаимодействовать с Рабочим столом, то ей придется предоставить права системы. С помощью этой вкладки можно также указать профиль оборудования, при использовании которого будет запускаться служба.

- Восстановление — с помощью данной вкладки можно определить действия, которые будет выполнять система в том случае, если службу не удалось запустить. При этом можно указать либо повторную попытку запуска службы, либо запуск другой программы, либо перезагрузку компьютера.
- Зависимости — с помощью этой вкладки можно определить службы, для работы которых необходима данная служба. При этом если вы отключите эту службу, то все остальные службы, которым она необходима для работы, также будут отключены. С помощью этой вкладки можно также определить службы, работа которых необходима для запуска службы. Как уже было сказано раньше, вкладка использует для поиска зависимых служб инструментарий WMI, поэтому если служба инструментария WMI отключена, то вкладка будет неактивна.

Глава 11

Оснастки администрирования Windows XP

- **Журналы и оповещения производительности**
- **Управляющий элемент WMI**
- **Просмотр событий**
- **Редактор объекта групповой политики**
- **Результирующая политика**
- **Шаблоны безопасности**

Журналы и оповещения производительности

Оснастка предназначена для наблюдения за работой устройств, установленных на компьютере. Она содержит большой набор функций, что является как несомненным плюсом, так и большим минусом. Минус заключается в довольно сложном механизме работы с оснасткой, который может сначала отпугнуть пользователя операционной системы Windows XP. Оснастка входит в состав стандартной консоли `perfmon.msc`, работа с которой и будет рассмотрена. Оснастка Журналы и оповещения производительности имеет GUID-номер `{7478EF61-8C46-11d1-8D99-00A0C913CAD4}`, после запрещения которого доступ к оснастке будет заблокирован. В консоль Производительность также входит ActiveX-объект Системный монитор, отображение которого можно запретить с помощью GUID-номера `{C96401CF-0E17-11D3-885B-00C04F72C717}`.

Чаще всего консоль Производительность используется для определения устройств компьютера, которые пора улучшить, достигнув тем самым максимального повышения производительности от покупки нового устройства. Именно с этой точки зрения и будет рассмотрена данная консоль.

Системный монитор

После ввода в командной строке Выполнить команды `perfmon.msc` консоль управления Microsoft обращается к содержимому ветвей реестра `HKEY_CURRENT_USER\Software\Microsoft\SystemMonitor` и `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Perflib`, параметры которых будут рассмотрены чуть позже. После ввода команды перед вами отобразится окно консоли Производительность, открытое на ActiveX-объекте Системный монитор (рис. 11.1).

С помощью ActiveX-объекта Системный монитор можно проследить работу какого-либо оборудования, установленного на компьютере, в реальном режиме времени. При этом существует множество параметров работы оборудования, которые можно просмотреть (с помощью так называемых счетчиков).

Окно системного монитора состоит из трех областей — области панели инструментов системного монитора (расположена вверху консоли и содержит набор кнопок быстрого доступа), области для отображения результатов работы устройств (средняя область) и области счетчиков, за которыми выполняется слежение (расположена внизу консоли).

Добавление счетчиков

Для примера попробуем добавить счетчики производительности. Для начала удалите все используемые в данный момент счетчики. Для этого нужно выделить счетчик в области счетчиков и нажать клавишу Delete. После этого вызовите контекстное меню в любой из областей системного монитора и выберите команду Добавить счетчики. Это приведет к отображению диалогового окна, приведенного на рис. 11.2.

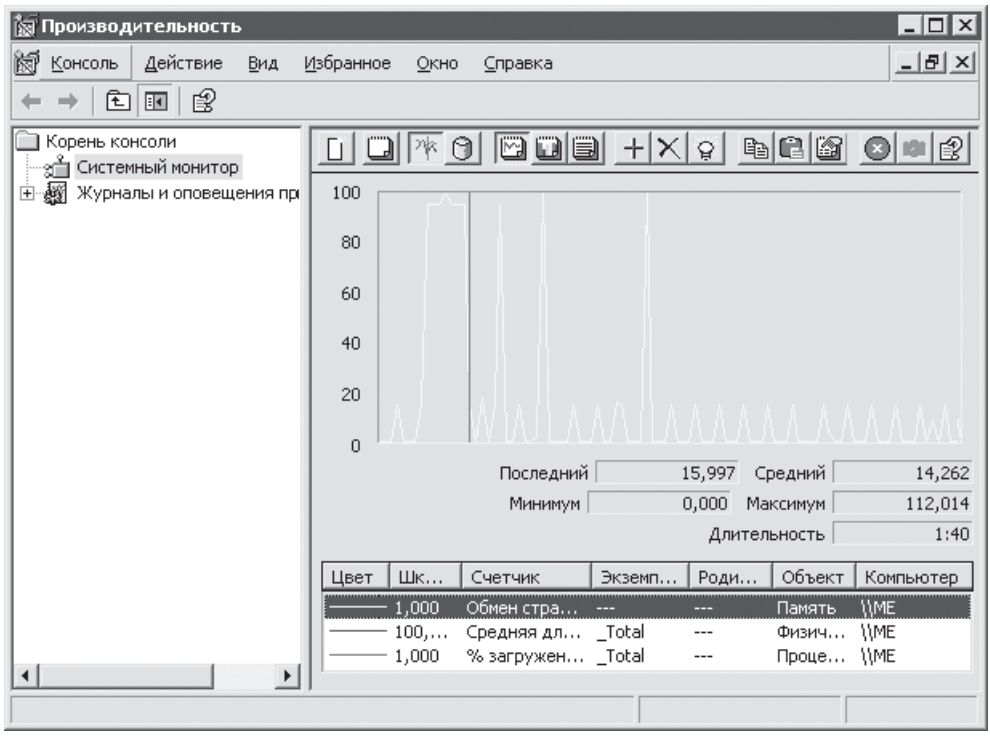


Рис. 11.1. Окно консоли Производительность

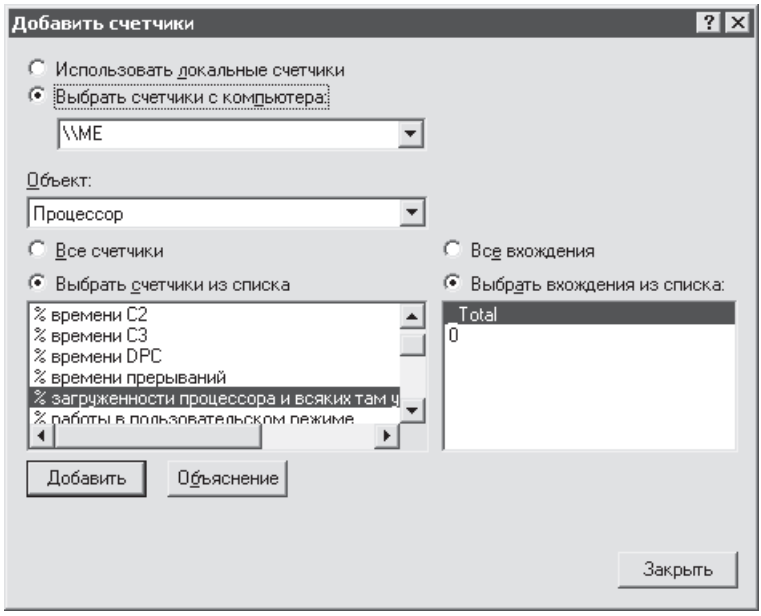


Рис. 11.2. Окно добавления счетчиков

С помощью данного окна можно определить, счетчики какого компьютера будут подключаться к консоли (локального или любого другого, подключенного к сети), определить устройство (список Объект), за работой которого вы будете следить, а также определить сами счетчики (переключатель Выбрать счетчики из списка) параметров работы устройства, которые будут отслеживаться. После выбора счетчика в правом окне можно выбрать экземпляр устройства, работа которого будет отслеживаться. На рис. 11.2 отображено только два экземпляра устройства — _Total и 0. Экземпляр 0 определяет первый процессор, установленный в системе, а экземпляр _Total определяет слежение за всеми процессорами, установленными на компьютере (при этом будет выводиться среднее арифметическое данных по работе процессоров). Если на компьютере установлен только один процессор, то экземпляр _Total эквивалентен экземпляру 0, но если бы компьютер содержал большее количество установленных процессоров, то присутствовали бы и другие экземпляры устройств: 1 — для второго процессора, 2 — для третьего процессора и т. д.

ПРИМЕЧАНИЕ

Если значение DWORD-параметра Disable Performance Counters, расположенного в ветви системного реестра HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Perflib, будет равно 1, то будет запрещено добавление счетчиков, а также работа с ActiveX-объектом Системный монитор. Если значение данного параметра будет равно 1, то список счетчиков просто-напросто будет пуст. Аналогичного результата можно добиться, если присвоить DWORD-параметру Updating той же ветви реестра значение, равное 0.

ВНИМАНИЕ

Название экземпляра _Total можно изменить на любое другое. Для этого применяется параметр строкового типа TotalInstanceName, расположенный в ветви реестра HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Perflib. При этом следует учитывать, что все счетчики, загруженные до изменения значения параметра TotalInstanceName, работать не будут.

Если просмотреть список объектов, за которыми можно следить, а также количество различных счетчиков, реализованных для них, можно ужаснуться. Что же из этого многообразия выбрать? Частично решить этот вопрос вам поможет кнопка Объяснение, после выбора счетчика и нажатия которой перед вами отобразится диалоговое окно с описанием того, за чем же следит данный счетчик. Вам также могут помочь советы профессионалов, которые предлагают следить за следующими компонентами (если необходимо определить общий уровень производительности компьютера или устройства, которые пора улучшить) компьютера: Процессор, Память, Система и Физический диск. Вкратце рассмотрим наиболее интересные счетчики данных устройств.

ПРИМЕЧАНИЕ

В любом случае, здесь не будут рассмотрены все объекты, счетчики которых можно использовать, так как количество объектов на различных компьютерах может быть разное. Это связано с тем, что любая служба может добавить свои собственные счетчики. Для этого достаточно в параметре строкового типа Library указать библиотеку, содержащую функции работы с новыми счетчиками. Параметр расположен в разделе Performance ветви реестра, хранящей сведения о данной службе (на страницах книги уже несколько раз упоминалось, что настройки служб находятся в отдельных разделах ветви реестра HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services). Можно также удалить возможность работы со счетчиками данного объекта, просто удалив раздел Performance (или переименовав его). Но можно поступить и намного проще — просто запретить работу счетчиков данного объекта. Для этого необходимо в разделе Performance данной службы создать DWORD-параметр Disable Performance Counters и присвоить ему значение 1.

Основные параметры раздела Performance неинтересны и содержат названия функций библиотеки для работы со счетчиками.

Для процессора это следующие счетчики.

- % загруженности процессора — в зависимости от экземпляра устройства определяет процент загруженности конкретного процессора или всех процессоров, установленных на компьютере. Аналогичную информацию можно просмотреть и в Диспетчере задач Windows (на вкладке Быстродействие). Если процессор постоянно загружен на 70–90 %, значит, пора покупать более мощный процессор.
- % времени прерываний — в зависимости от экземпляра устройства определяет процент времени загруженности конкретного процессора или всех процессоров, установленных на компьютере, в течение которого процессор обрабатывает различные аппаратные прерывания. Если показания данного счетчика колеблются в пределах 25–35 %, стоит также подумать о более мощном процессоре.

Для памяти это следующие счетчики.

- Обмен страниц в секунду — определяет количество страниц, считываемых или записываемых на диск в течение одной секунды. Как правило, страницы записываются на диск в том случае, если оперативная память компьютера исчерпана и системе приходится использовать файл подкачки. Иными словами, чем меньше показания данного счетчика, тем лучше. В случае большого объема обмена страниц (имеется в виду обмен страницами в течение всего сеанса работы пользователя, а не временный обмен, который может быть ассоциирован записью информации на диск (а следовательно, не отображает реальную картину загруженности памяти)) рекомендуется купить дополнительную планку оперативной памяти.
- Ошибок страницы в секунду — указывает количество ошибок доступа к оперативной памяти, возникающих при отсутствии в оперативной памяти необходимых

данных. Как правило, после выявления ошибки системе приходится обращаться к содержимому жесткого диска за получением необходимой информации и повторного помещения ее в память, что занимает большое количество времени (сравнительно). При показаниях данного счетчика, превышающих значение 5, рекомендуется купить дополнительные планки оперативной памяти.

- Доступно байт — определяет количество свободной в данный момент виртуальной памяти. Если показания этого счетчика постоянно колеблются в пределах 10–20 Мбайт, рекомендуется купить дополнительные планки оперативной памяти.

Для физического диска это следующие счетчики.

- % активности диска — определяет процент времени, которое жесткий диск тратит на удовлетворение запросов на чтение/запись данных. Если показания данного счетчика долгое время колеблются в районе 80–100 %, то необходимо подумать над покупкой более быстрого жесткого диска или дополнительного объема оперативной памяти.
- Текущая длина очереди диска — указывает количество запросов на выполнение чтение/запись на диск, ожидающих своего выполнения в очереди. Показания счетчика, отображающие больше двух запросов в очереди, уже считаются поводом покупки дополнительной оперативной памяти или более быстрого жесткого диска.

Для системы это следующий счетчик.

Длина очереди процессора — определяет количество процессов, ожидающих своего выполнения в очереди процессов. Показания счетчика, отображающие больше двух процессов, уже считаются поводом покупки более производительного процессора.

ПРИМЕЧАНИЕ

Названия счетчиков, а также описания их работы на разных компьютерах могут отличаться. Это связано с тем, что сведения о названиях счетчиков и их описания хранятся не в файле библиотеки, а непосредственно в реестре. Для их хранения применяются два параметра REG_MULTI_SZ-типа ветви реестра HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Perflib\019 (для англоязычной версии операционной системы используется конечный подраздел 009). Данная ветвь реестра содержит два параметра — Counter и Help. Первый из них определяет названия счетчиков, а второй — их описание.

Для примера загрузим в консоль следующие счетчики: % времени прерываний и Длина очереди процессора для процессора, Ошибок страниц в секунду для памяти, % активности дисков и Текущая длина очереди диска для физического диска. Чтобы загрузить счетчик, необходимо выделить его и нажать кнопку Добавить. После этого

счетчик загрузится, но диалог Добавить счетчики закрыт не будет. После того, как вы добавите все необходимые счетчики, просто нажмите кнопку Закрыть, чтобы перейти к окну консоли.

Изменение вида системного монитора

После того как вы нажмете кнопку Закрыть в диалоге Добавить счетчики, перед вами предстанет консоль Производительность, в которую будут загружены все указанные вами счетчики. При этом по умолчанию будет использоваться способ отображения показаний счетчика в виде графика (рис. 11.3).

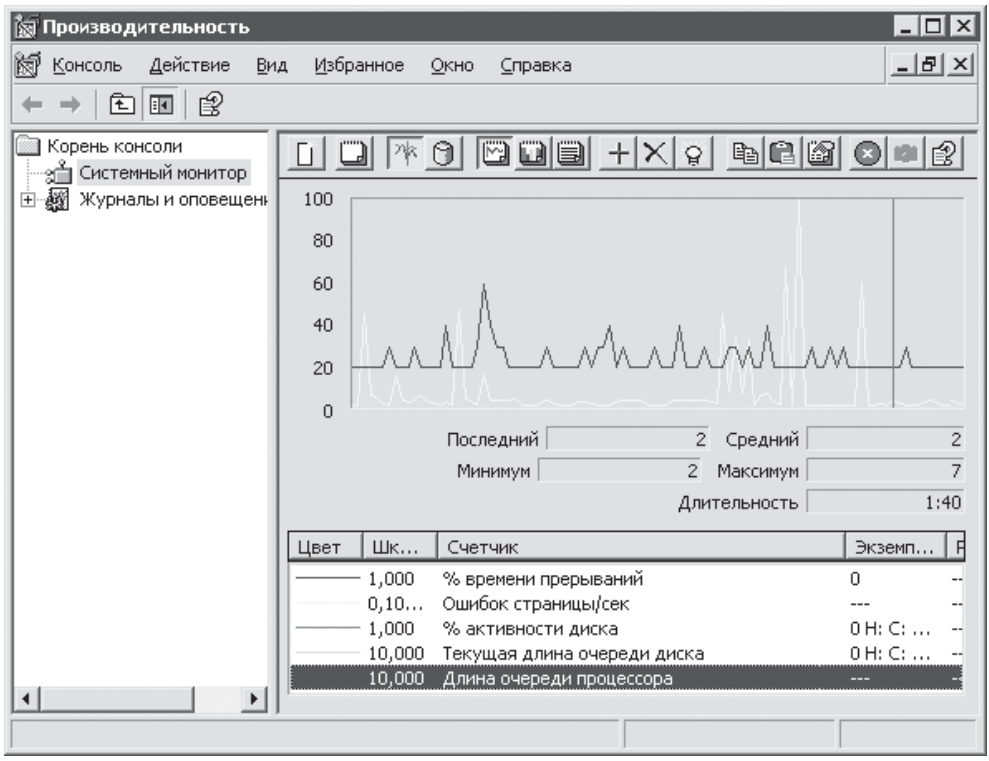


Рис. 11.3. Способ отображения показаний счетчиков в виде графика

При просмотре небольшого количества счетчиков этот способ отображения наиболее оптимален. Заметьте, что под графиком отображаются поля Последний, Средний, Минимум, Максимум и Длительность, в которых приведены общие показания выделенного в данный момент счетчика. Но при просмотре показаний большого количества счетчиков, да еще и с огромным разбросом по шкале графика, слежение за показаниями может быть затруднительно. Поэтому существует возможность применения гистограммы или отчета показаний счетчиков вместо графика. Для изменения вида показаний счетчиков используются три кнопки панели инструментов системного монитора, представленные в табл. 11.1.

Таблица 11.1. Кнопки изменения вида показаний счетчиков

Изображение кнопки	Вид показаний счетчиков
	Отобразить показания счетчиков в виде отчета. Вид может быть оптимален при просмотре показаний большого количества счетчиков, описывающих большое количество оборудования, расположенного на разных компьютерах сети
	Отобразить показания счетчиков в виде гистограммы. Вид может быть оптимален при просмотре показаний большого количества счетчиков, описывающих большое количество оборудования, расположенного на локальном компьютере
	Отобразить показания счетчиков в виде графика. Вид может быть оптимален при просмотре показаний небольшого количества счетчиков

Существует также возможность изменения цвета, которым отображается конкретный счетчик. Для этого необходимо в контекстном меню данного счетчика выбрать команду Свойства. После этого будет открыто диалоговое окно Свойства: Системный монитор на вкладке Данные. Вкладка содержит список Цвет, с помощью которого изменяется цвет отображения данного счетчика.

Еще одной интересной вкладкой диалога Свойства: Системный монитор является вкладка Источник, с помощью которой можно определить источник показаний счетчиков, отображаемых системным монитором. По умолчанию используются текущие показания счетчиков, но существует возможность загрузки в системный монитор показаний, описанных в файле журнала или базе данных. Вероятно, пока что данные возможности системного монитора вам будут непонятны, поэтому рассматривать их не станем. Вместо этого мы закончим рассказ об ActiveX-объекте Системный монитор и перейдем к рассмотрению оснастки Журналы и оповещения производительности, ведь именно с ее помощью создаются файлы журналов показаний счетчиков или базы данных SQL, которые и используются на вкладке Источник диалога Свойства: Системный монитор как возможные источники показаний счетчиков для работы системного монитора.

Журналы и оповещения производительности

Несмотря на то, что просмотр счетчиков в реальном времени является хорошим способом определения производительности компьютера, он имеет ряд недостатков. Главным из них является то, что при просмотре счетчиков пользователь, как правило, больше ничего на компьютере не делает (не играет, не печатает, то есть компьютер просто простаивает), поэтому некоторые из счетчиков в этот момент могут быть просто неактуальны. Решить эту проблему можно с помощью оснастки Журналы и оповещения производительности. Благодаря этой оснастке можно настроить такие функции компьютера, как возможность ведения журналов счетчиков, журналов трассировки и оповещения о каком-либо событии.

Журналы счетчиков

Именно с помощью журналов счетчиков решается проблема просмотра счетчиков в реальном времени. С помощью данных журналов можно определить время, начиная с которого компьютер будет записывать показания счетчиков в журнал, а также время, после которого запись в журнал будет прекращена. После этого можно продолжить повседневную работу с компьютером, а уже в конце дня просто загрузить созданный журнал счетчиков в системный монитор, чтобы просмотреть их показания.

По умолчанию существует уже созданный журнал счетчиков, называемый Обзор системы. Это пример журнала, на основе которого можно определить саму суть создания журнала счетчиков. Его можно запустить или остановить, то есть данный журнал является полнофункциональным, но его изменение с помощью консоли Производительность запрещено, поэтому ниже будут рассмотрены некоторые параметры реестра, с помощью которых можно отредактировать данный журнал.

Для примера можно создать свой собственный журнал счетчиков.

ПРИМЕЧАНИЕ

Информация обо всех журналах счетчиков хранится в реестре. Для этого предназначена ветвь реестра HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\SysmonLog\Log Queries. Она принадлежит службе Журналы и оповещения производительности, запускаемой в качестве сетевой службы. Другими словами, если данная служба будет остановлена, то нельзя будет работать с журналами счетчиков. Каждый журнал счетчиков создает в приведенной ветви реестра свой собственный раздел, имеющий название в формате GUID-номера. Например, для журнала счетчиков Обзор системы используется раздел {123a660c-c5ce-469a-ad49-7dee9de9376c}.

Для создания нового журнала необходимо в контекстном меню элемента Журналы счетчиков выбрать команду Новые параметры журнала. После этого перед вами отобразится диалоговое окно для ввода имени журнала, а затем появится окно параметров журнала, изображенное на рис. 11.4.

С помощью вкладки Общие можно добавить счетчики, показания которых будут заноситься в журнал производительности (кнопка Добавить счетчики), определить интервал времени, с которым показания счетчиков будут сниматься и заноситься в журнал (область Снимать показания каждые:), а также указать учетную запись пользователя, от имени которого будет запускаться данный журнал (поле От имени). Работа с диалогом, отображаемым после нажатия кнопки Добавить счетчики, ничем не отличается от работы с уже рассмотренным диалогом Добавить счетчики. Для добавления счетчиков можно также воспользоваться кнопкой Добавить объекты. В этом случае будут добавлены все счетчики какого-либо объекта. Стоит также взглянуть на поле Текущий файл журнала. На данной вкладке его запрещено редактировать, хотя на других вкладках можно будет отредактировать как путь к журналу, так и его имя. Тем не менее благодаря реестру существует еще одна интересная

возможность настройки создания журналов — определение пути к папке, в которую будут помещаться журналы счетчиков по умолчанию. Если вам для всех создаваемых журналов приходится определять другой каталог хранения (по умолчанию для хранения журналов счетчиков используется системный диск), то предлагаю воспользоваться параметром строкового типа `DefaultLogFileFolder`, расположенным в ветви реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\SysmonLog`.

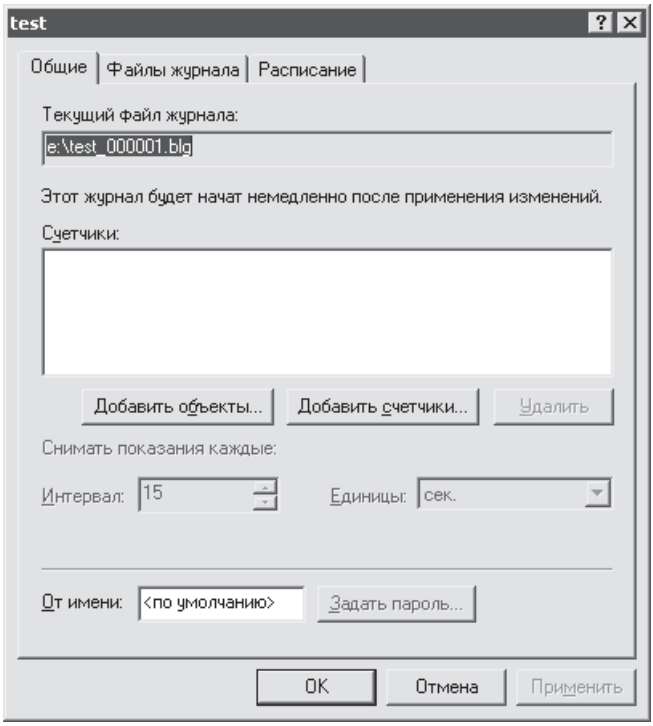


Рис. 11.4. Параметры создания нового журнала счетчиков

На вкладке Файлы журнала (пока вы не укажете хотя бы один счетчик для снятия показаний, вам будет запрещено переходить на другие вкладки диалога параметров) можно настроить сам журнал счетчиков. Главным образом можно настроить его имя и каталог для хранения (в диалоге, вызываемом после нажатия кнопки Настроить), а также определить индекс, добавляемый к создаваемым файлам (флажок Имена файлов оканчиваются на). На данной вкладке можно также определить тип создаваемого журнала. Возможны следующие типы.

- Двоичный файл — используется по умолчанию и является оптимальным способом создания файлов журналов, если их необходимо просматривать только в системном мониторе.
- Двоичный циклический файл — отличается от предыдущего лишь тем, что при достижении конца файла журнала его содержимое будет перезаписываться заново.

- Текстовый файл — существует два вида текстовых файлов: разделитель — запятая и разделитель — табуляция. Различия между ними описаны в самих названиях файлов. Плюсом текстовых файлов является возможность просмотра их содержимого с помощью таких программ, как Microsoft Excel или даже Блокнот. Минусом же является меньшая скорость обработки данных при просмотре содержимого файлов журналов.
- База данных SQL — позволяет заносить показания счетчиков в общую базу данных SQL.

На вкладке Расписание можно задать способ запуска и остановки слежения за показаниями счетчиков, а также команду, которая будет выполняться при остановке снятия показаний счетчиков. При этом можно указать время запуска и остановки либо указать запуск и остановку счетчиков вручную. Если будет выбран режим Вручную, то для запуска и остановки снятия показаний счетчиков необходимо будет воспользоваться контекстным меню созданного вами журнала счетчиков, выбрав, соответственно, команду Запуск или Остановка (не самого файла журнала счетчиков, а созданного элемента журнала счетчиков в консоли Производительность).

СОВЕТ

Следует всегда внимательно относиться к возможности запуска различных программ после выполнения конкретных действий. Особенно в том случае, если программа будет запускаться от имени администратора. По крайней мере, запускаемая программа всегда должна находиться в каталоге, изменение содержимого которого запрещено пользователям системы, иначе ничто им не помешает заменить файл запускаемой программы любым другим, реализовав тем самым метод взлома системы через посредника.

ПРИМЕЧАНИЕ

Создать журнал счетчиков можно и с помощью командной строки. Для этого применяется команда logman create counter. На страницах книги работа с этой командой описана не будет, тем не менее она не представляет трудности, если вы знаете, как создаются журналы счетчиков с помощью консоли Производительность. Для просмотра параметров данной команды введите в командной строке команду logman create counter /?.

После того как вы создадите журнал счетчиков, он отобразится в том же списке, что и журнал Обзор системы. При этом значок напротив журнала счетчиков будет красного цвета, это говорит о том, что в данный момент журнал остановлен. После запуска журнала счетчиков значок напротив него станет зеленого цвета. Но допустим, что у нас уже есть файл показаний счетчика, то есть наш журнал счетчиков был запущен и остановлен (файл показаний счетчиков должен содержать больше двух показаний, иначе его нельзя будет использовать, то есть по умолчанию показания должны сниматься как минимум 45 секунд). Что же теперь делать с созданным файлом показаний счетчиков? Во-первых, можно указать путь к нему

на вкладке Источник рассмотренного диалога Свойства: Системный монитор. А можно воспользоваться командой Сохранить параметры как из контекстного меню созданного журнала счетчиков. С помощью данной команды можно будет создать HTML-файл, хранящий ActiveX-объект Системный монитор. Запуск этого HTML-файла приведет к появлению уже знакомого окна Системный монитор, которое по умолчанию будет использовать для своей работы показания счетчиков, хранящиеся в созданном с помощью данного журнала счетчиков файле показаний.

Теперь рассмотрим параметры реестра, которые используются для хранения параметров журналов счетчиков. Как было сказано выше, все журналы счетчиков имеют свой собственный раздел в ветви реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\SysmonLog\Log Queries`. Все разделы данной ветви реестра представляют собой CLSID-номера, генерируемые системой при создании журналов счетчиков. Каждый раздел, соответствующий журналу счетчиков, может содержать следующие параметры, которые можно редактировать даже для журнала счетчиков Обзор системы, несмотря на то, что изменение параметров этого журнала с помощью оснастки невозможно.

- `Collection Name` — параметр строкового типа, определяет имя журнала счетчиков. Значение переопределяется параметром строкового типа `Collection Name Indirect`.
- `Counter List` — этот параметр `REG_MULTI_SZ`-типа определяет названия счетчиков, показания которых будут считываться.
- `Create New File` — если значение данного параметра `DWORD`-типа равно `1`, то при следующем запуске журнала счетчиков будет создан новый файл (а не переписан уже существующий).
- `Current Log File Name` — этот параметр строкового типа определяет путь к файлу (и его имя), в который будут записываться показания счетчиков. При этом значение состоит из значений двух других параметров строкового типа — `Log File Folder` и `Log File Base Name`. Первый из них определяет путь к каталогу, содержащему файл счетчиков, а второй параметр определяет название файла.
- `Log File Max Size` — параметр `DWORD`-типа, определяет максимальный размер создаваемого файла.

Журналы трассировки

Журналы трассировки являются разновидностью журналов счетчиков (более того, они описываются в той же ветви реестра), собирающей наиболее полные сведения о тех или иных объектах системы. Применять журналы трассировки, как правило, следует только при тестировании того или иного объекта системы, так как их ведение требует от компьютера большого объема системных ресурсов.

По умолчанию не существует ни одного примера создания журнала трассировки, поэтому попробуем создать свой собственный журнал. Для этого необходимо из контекстного меню элемента Журнал трассировки выбрать команду Новые параметры журнала. После этого система попросит ввести имя создаваемого журнала трас-

сировки (так как журналы трассировки хранятся в той же ветви реестра, что и журналы счетчиков, запрещено давать новым журналам трассировки имена, уже используемые журналами счетчиков). После ввода имени перед вами отобразится диалоговое окно параметров нового журнала трассировки. Скажем сразу, что данный диалог включает в себя вкладки Файлы журнала и Расписание, содержимое которых аналогично содержимому данных вкладок для создания журнала счетчиков, поэтому рассмотрены они не будут.

ПРИМЕЧАНИЕ

Если при создании нового журнала счетчиков на вкладке Расписание по умолчанию указано, что данный журнал запускается вручную, то после создании журнала трассировки он будет запускаться ежедневно в то время, когда вы его создали.

Диалог параметров нового журнала трассировки содержит вкладку Общие. С ее помощью можно указать поставщиков возможности трассировки (чем-то напоминают несколько совмещенных вместе объектов журналов счетчиков). При этом существует возможность использования как системных поставщиков, так и поставщиков, устанавливаемых вместе с дополнительными службами. Чтобы выбрать системного поставщика, нужно установить переключатель в положение События, протоколируемые системным поставщиком. После этого станет активным ряд флажков, с помощью которых можно указать те из событий, протоколирование которых будет выполняться.

ПРИМЕЧАНИЕ

Как правило, журналы трассировки могут запускаться только от имени администратора, поэтому на вкладке Общие необходимо указать запуск от имени администратора и пароль для запуска.

В диалоге параметров журнала трассировки присутствует дополнительная вкладка, которая называется Дополнительно. С ее помощью можно определить размер буферов трассировки и их количество. Все дело в том, что данные трассировки сначала записываются в буфер трассировки, а потом уже, когда буфер будет заполнен, — в файл трассировки.

Но чем же, с точки зрения реестра, отличаются журналы трассировки от журналов счетчиков? Как оказалось, они отличаются значением всего одного параметра. Если значение DWORD-параметра `Log Type` равно `0`, то данный журнал является журналом счетчиков, а если значение параметра `Log Type` равно `1`, то журналом трассировки. Например, если значение параметра `Log Type` ветви реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\SysmonLog\Log Queries\{123a660c-c5ce-469a-ad49-7dee9de9376c}` будет равно `1`, то стандартный журнал Обзор системы станет журналом трассировки, а не журналом счетчиков.

ПРИМЕЧАНИЕ

Если значение параметра Log Type равно 0xffffffff, то журнал будет скрыт.

Для журналов трассировки могут применяться следующие дополнительные параметры DWORD-типа.

- `Trace Buffer Flush Interval` — определяет интервал времени в секундах, с которым будет выполняться сбрасывание содержимого буферов в файл трассировки.
- `Trace Buffer Min Count` — указывает минимальное количество буферов трассировки, которое будет использоваться в любом случае.
- `Trace Buffer Max Count` — определяет максимальное количество буферов трассировки. Если количества буферов, определенного в параметре `Trace Buffer Min Count`, не хватает для буферизации данных, то будут созданы дополнительные буферы (общее количество буферов не должно быть больше, чем значение параметра `Trace Buffer Max Count`).
- `Trace Buffer Size` — указывает размер в килобайтах каждого буфера трассировки.

Оповещения

Оповещения — это еще один механизм, работающий на основе показаний счетчиков. С его помощью можно определить счетчики, показания которых будут сниматься при работе оповещения, а также определить пороги, преодоление которых счетчиками вызовет какое-либо действие.

По умолчанию не существует ни одного оповещения, поэтому, чтобы создать новое оповещение, необходимо воспользоваться контекстным меню элемента Оповещения. После выбора команды Новые параметры оповещений консоль управления Microsoft попросит ввести имя нового оповещения, после чего перед вами отобразится диалоговое окно настройки параметров оповещения.

Окно содержит знакомые вкладки Общие, Действие и Расписание. На вкладке Общие можно указать счетчики, показания которых будут считываться, а также порог, после преодоления которого произойдут события, указанные на вкладке Действия. На вкладке Действия можно определить, будет ли происходить запись в журнал приложений (оснастка Просмотр событий) при возникновении оповещения, будет ли запускаться один из журналов трассировки или счетчиков, а также определить команду, которая будет выполняться при возникновении оповещения. На вкладке Расписание можно задать время, начиная с которого будет запускаться данное оповещение. По умолчанию оповещение будет запускаться сразу же после своего создания.

Настройки оповещений также находятся в разделах ветви системного реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\SysmonLog\Log Queries`. При этом значение DWORD-параметра `Log Type`, равное 2, как раз и определяет, что данная ветвь реестра описывает оповещение.

Управляющий элемент WMI

Оснастка Управляющий элемент WMI включает в себя настройки инструментария управления Windows (WMI), с помощью которого можно удаленно или локально управлять различными настройками операционной системы Windows. WMI реализовано на основе протокола WBEM (протокол управления предприятием на основе веб-технологий) и включает в себя CIM-совместимую базу данных (CIMOM), содержащую сведения об объектах системы, а также диспетчер CIM, с помощью которого реализованы функции работы с поставщиками WMI. Поставщики WMI являются посредниками между WMI и компонентами компьютера. Именно с их помощью реализуются такие возможности, как считывание, оповещение о событии и изменение данных состояния компонентов компьютера. Сама же возможность WMI может применяться как в программировании под Windows с помощью WinAPI, так и при создании сценариев сервера сценариев Windows. Возможности WMI также используются в компонентах Свойства системы, Сведения о системе и для формирования вкладки Зависимости, находящейся в окне диалога свойств конкретной службы в оснастке Службы.

ПРИМЕЧАНИЕ

Именно база данных CIMOM содержит всю информацию о компонентах компьютера, установленных на компьютере программах и многом другом. Информация в базе CIMOM компьютера обновляется при каждом входе в систему или подключении к WMI.

Свойства WMI

Для работы оснастки Управляющий элемент WMI необходимо, чтобы в системе был зарегистрирован GUID-номер {5C659257-E236-11D2-8899-00104B2AFB46}. Именно этот GUID-номер и идентифицирует настройки оснастки Управляющий элемент WMI. Эта оснастка входит в состав консоли Инфрастуктура управления WMI, открыть которую можно с помощью команды wmimgmt.msc. После ее ввода перед вами отобразится пустое окно консоли, содержащее единственный элемент дерева консоли — Элемент управления WMI (локальный). Контекстное меню данного элемента включает в себя две основные команды: Подключение к другому компьютеру и Свойства. Первая предназначена для просмотра WMI удаленного компьютера, а вторая позволяет просмотреть настройки WMI локального компьютера. После выбора команды Свойства перед вами отобразится диалоговое окно, подобное приведенному на рис. 11.5.

На вкладке Общие диалога свойств отображается общая информация о компьютере, к WMI которого вы подключились. С помощью данной вкладки можно изменить учетную запись, от имени которой вы подключились. Для этого предназначена кнопка Изменить. На вкладке Ведение журнала можно определить путь к журналу событий WMI, его размер, а также сведения, которые будут помещаться в этот журнал. На вкладке Архивация или восстановление можно вручную выполнить такие операции, как архивация или восстановление CIM-совместимой базы данных

WMI. На вкладке Дополнительно можно определить пространство имен WMI, используемое по умолчанию при разработке сценариев (если конкретное пространство имен указано не было). На вкладке Безопасность можно определить права доступа для различных пространств имен WMI. По умолчанию администраторы имеют полный доступ ко всему пространству имен, а остальным группам пользователей разрешен только доступ на выполнение методов пространства имен.

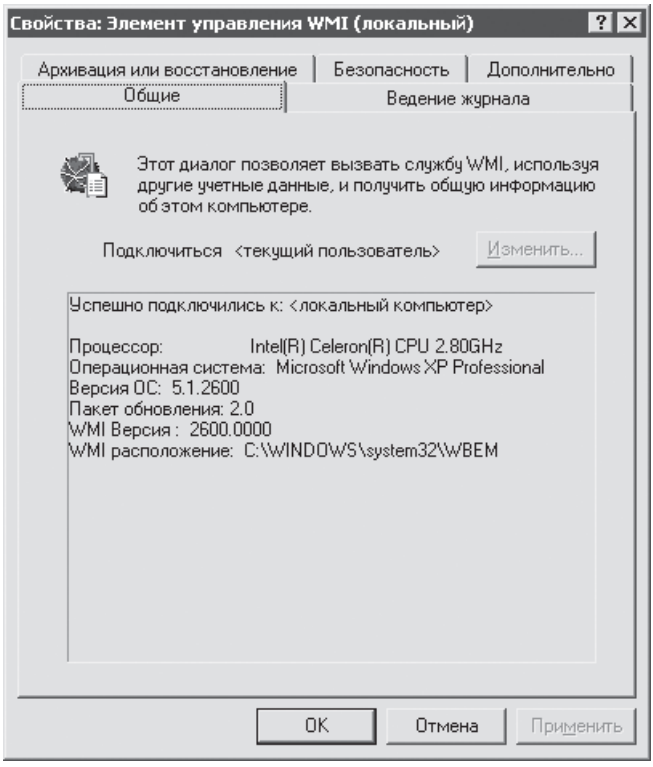


Рис. 11.5. Свойства WMI

Если читатель знаком с программированием на языке C++, то ему известен термин Пространство имен. Это логическое объединение различных функций, переменных, классов и т. д., направленное на улучшение структурированности кода и исключение конфликтов между функциями, имеющими одинаковые названия. В контексте WMI с помощью пространства имен реализована система безопасности. Другими словами, если пользователю запрещен доступ к одному из пространств имен, то он не сможет использовать функции, которые были описаны в этом пространстве.

Настройки WMI в реестре

В предыдущем разделе вкратце были рассмотрены настройки WMI, которые можно изменить с помощью диалога Свойства: Элемент управления WMI (локальный).

Теперь же рассмотрим настройки WMI, доступ к которым предоставляет реестр Windows XP. Все эти настройки содержатся в ветви HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\WBEM в следующих параметрах строкового типа.

- Build — определяет номер версии WMI, установленной в системе. По умолчанию в Windows XP используется версия 2600.0000.
- Installation Directory — указывает путь к каталогу, хранящему файлы WMI. В этом каталоге расположены все библиотеки, необходимые для работы WMI, файлы журналов и многое другое.
- MOF Self-Install Directory — определяет путь к каталогу, в который будут помещаться файлы с расширением MOF, предназначенные для автоматического добавления новых поставщиков WMI, а также функций и методов.

Кроме параметров, данная ветвь реестра включает в себя вложенный раздел CIMOM, содержащий сведения о CIM-совместимой базе данных WMI. Этот раздел хранит следующие параметры.

- Autorecover MOFs — данный параметр REG_MULTI_SZ-типа включает в себя список всех файлов с расширением MOF, используемых при инициализации и восстановлении базы данных WMI. Содержимое этого файла также определяет порядок, в котором компилировались файлы MOF при установке WMI.
- Backup Interval Threshold — параметр строкового типа, определяет промежуток времени в минутах, по истечении которого будет выполняться резервное копирование базы данных WMI. По умолчанию в реестре не существует.
- EnableEvents — этот параметр строкового типа определяет, будет ли использоваться подсистема событий WMI. Если значение равно 1, то будет. Как правило, значение параметра равно 0.
- EnableStartupHeapPreallocation — параметр DWORD-типа, определяет, будет ли сразу же при запуске WMI выделяться куча (часть оперативной памяти, используемая для хранения объектов). Если значение равно 1, то при запуске WMI будет заранее выделяться куча, размер которой определен в параметре LastStartupHeapPreallocation. По умолчанию эти параметры не существуют.
- EnablePrivateObjectHeap — этот параметр DWORD-типа определяет, будет ли использоваться куча для хранения объектов от поставщиков. По умолчанию не существует.
- EnableObjectValidation — если значение этого параметра DWORD-типа будет равно 1, то проверка целостности объектов от поставщиков будет выполняться. По умолчанию в реестре не существует.
- High Threshold On Client Objects (B) — этот параметр DWORD-типа определяет верхнее пороговое значение очереди объектов от поставщиков, достижение которого приводит к прекращению приема объектов от поставщиков (в этом случае WMI возвращает поставщикам код WBEM_E_OUT_OF_MEMORY). По умолчанию параметр не существует.

- `High Threshold On Events (B)` — значение этого параметра `DWORD`-типа аналогично значению предыдущего, но в этом случае определяется верхнее пороговое значение очереди событий (а не объектов) от поставщиков.
- `Log File Max Size` — определяет максимальный размер файлов журналов, создаваемых службами WMI.
- `Logging` — этот параметр строкового типа определяет уровень протоколирования ошибок и может принимать следующие значения:
  - `0` — отключить протоколирование;
  - `1` — краткое протоколирование ошибок;
  - `2` — полное протоколирование ошибок.
- `Logging Directory` — этот параметр строкового типа содержит путь к каталогу, в котором находятся файлы системных журналов WMI. Именно значение этого параметра и редактируется на вкладке Ведение журнала.
- `Low Threshold On Client Objects (B)` — параметр `DWORD`-типа, определяет нижнее пороговое значение очереди объектов от поставщиков, достижение которого приводит к замедлению скорости создания объектов. По умолчанию в реестре не существует.
- `Low Threshold On Events (B)` — этот параметр `DWORD`-типа определяет нижнее пороговое значение очереди событий от поставщиков, достижение которого приводит к замедлению скорости создания событий.
- `Max DB Size` — параметр строкового типа, определяет максимальный размер базы данных WMI. По умолчанию не существует.
- `Max Wait On Events (ms)` — этот параметр строкового типа указывает время в миллисекундах, в течение которого событие может находиться в очереди. Если по истечении этого времени событие все еще находится в очереди, то оно будет автоматически удалено.
- `Max Wait On Client Objects (ms)` — параметр строкового типа, указывает время в миллисекундах, в течение которого объект может находиться в очереди. Если по истечении этого времени объект все еще находится в очереди, то он будет автоматически удален. По умолчанию в реестре не существует.
- `Repository Directory` — этот параметр строкового типа определяет путь к каталогу, используемому службой WMI для хранения архивов CIM-совместимой базы данных. Они используются при восстановлении базы данных.
- `Working Directory` — параметр строкового типа, определяет путь к рабочему каталогу WMI. По умолчанию используется путь `%systemroot%\system32\wbem`.

В реестре также существует ветвь `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\WBEM\Scripting`. Значения данной ветви определяют настройки взаимодействия WMI и сервера сценариев Windows. Например, ветвь может содержать следующие параметры.

- `Default Namespace` — этот параметр строкового типа определяет пространство имен, используемое по умолчанию (если пространство имен не указано явно). Чаще всего значение равно `root\cimv2`.
- `Enable for ASP` — если значение данного параметра `DWORD`-типа равно `1`, то будет разрешено использование сценариев WMI для ASP.
- `Default Impersonation Level` — этот параметр `DWORD`-типа определяет используемый по умолчанию (если уровень не указан в сценарии) уровень прав безопасности. По умолчанию значение равно `3`.

Доступ к WMI с помощью сервера сценариев

Полное понимание WMI невозможно без знания тех возможностей, которые она предоставляет администратору. Поэтому рассмотрим сейчас некоторые примеры написания сценариев сервера сценариев Windows с использованием возможностей WMI. Раздел не описывает работу с сервером сценариев — предполагается, что вы уже знаете, что это такое и как писать простые сценарии на языках VBScript или JScript. Здесь будет дано несколько примеров сценариев, на основе которых можно представить, какие возможности открываются перед администратором при использовании WMI, а также будет дана ссылка на один интересный каталог файловой системы Windows, содержащий список всех функций, реализованных в пространствах имен, и краткое описание этих функций.

Итак, сервер сценариев Windows является объектно-ориентированным языком. Иными словами, при написании сценариев в них можно подключать различные объекты (хранящие описания методов, реализующих различные возможности работы с операционной системой), а также моникеры (COM-механизм для обеспечения привязки к COM-объекту, например к базе данных WMI). Сценарии сервера сценариев можно писать на языке VBScript либо Jscript. В данном случае для рассмотрения примеров будет использоваться язык VBScript.

Есть два способа запуска сценариев сервера сценариев Windows — либо с помощью двойного щелчка кнопкой мыши на файле (или с помощью команды `wscript.exe`), либо с помощью команды `cscript.exe`. Программа `cscript.exe` предназначена для работы со сценарием из командной строки, и ее плюсом является возможность указания параметров работы сценария (если он обрабатывает параметры). Приведенные ниже примеры, как правило, используют параметры командной строки, поэтому для их применения лучше обратиться к программе `cscript.exe`.

Включение и выключение SystemRestore для отдельных дисков

Рассмотрим первый пример тех возможностей, которые предоставляет администратору WMI. В этом примере воспользуемся классом `SystemRestore`, описанным в пространстве имен `root/default` для реализации возможности отключения или включения восстановления системы на отдельных дисках с помощью сервера сценариев Windows. Пример, кроме доступа к WMI, будет содержать

описание реализации основных возможностей объектов сервера сценариев, а также подробное объяснение, для чего записывается та или иная строка сценария.

11.1.

```
'SystemRestore.

set objFS = CreateObject("Scripting.FIleSystemObject")

set objTextFile = objFS.OpenTextFile("d:\sr_log.txt", 8, True)

set objREG = WScript.CreateObject("Wscript.Shell")

Set Args = wscript.Arguments
```

```
If Args.Count() > 0 Then
    Drive = Args.item(0)
Else
    Drive = "c:\"
End If

'[illegible]

If Args.Count() > 1 Then
    StopSR = Args.item(1)
Else
    StopSR = "N"
End If

'[illegible]
'[illegible]
'[illegible] – [illegible]
'[illegible] Write, [illegible]
'[illegible]
'[illegible]

objTextFile.Write Date() & " [illegible] " & Time()

'[illegible]
'[illegible]
'[illegible] Windows

call sronoff()

'[illegible] ([illegible] sronoff).
'[illegible] SystemRestore [illegible]
'[illegible]
'[illegible]
'[illegible]
'[illegible]
'[illegible]

private sub sronoff()

'[illegible] WMI [illegible]
winmgmts.
'[illegible] root/default [illegible]
'[illegible] SystemRestore [illegible] c [illegible]
```

```
'                    (impersonate).
'     root/default,                                SystemRestore
'                                                                    ,
'                                                WMI
'
```

```
Set obj = GetObject("winmgmts:{impersonationLevel=impersonate}!root/
default:SystemRestore")
```

```
'                                                                    ,
'                                                                    ,
'                                                                    ,
'                                                                    ,
'                                                                    ,
'                                                                    ,
'
```

```
ON ERROR RESUME NEXT
```

```
'
'           DisableSR                                                1,
'    SystemRestore                                                   .
'                                                                    ,
'
```

```
dasableSR = objREG.RegRead("HKLM\SOFTWARE\Microsoft\Windows NT\
CurrentVersion\SystemRestore\DisableSR")
```

```
'                                                                    ,
'                                                                    ,
'                                                                    er
'                                -2147024894(                        .
'                                                                    ,
'                                                                    )
'                                                SystemRestore       ,
'                                                                    /
(                                                                    )
'                                                                    ,
'                                                                    ,
'                                                Windows
'                                                                    ,
'
```

```
if (err.Number = -2147024894) and (Drive <> "c:\") then
   objTextFile.WriteLine ": DisableSR                                ,
SystemRestore                                                        .
"
```

```
'[illegible], [illegible], [illegible]
'[illegible], [illegible]
'[illegible], [illegible]/[illegible]
'[illegible], [illegible] SystemRestore [illegible].

exit sub
End if

'[illegible] C:\ [illegible], [illegible]
'[illegible] \ [illegible]
'[illegible] SystemRestore [illegible], [illegible]
'SystemRestore [illegible] SystemRestore [illegible]
'[illegible] DisableSR [illegible]
'[illegible] 1. [illegible]/[illegible] SystemRestore
'[illegible]
'[illegible] DisableSR, [illegible] 1 ([illegible]). [illegible]
'[illegible] – [illegible] SystemRestore [illegible]
'[illegible], [illegible] DisableSR [illegible] 0, [illegible],
'[illegible]
'[illegible]

if (dasableSR > 0) and (Drive <> "c:\") then
   objTextFile.WriteLine " : [illegible] SystemRestore [illegible]
[illegible]"
exit sub
else

'[illegible]
'If..else [illegible] select case [illegible]
'[illegible], [illegible]

select case StopSR
case "N"
If (obj.Disable(Drive)) = 0 Then
   objTextFile.WriteLine " : SystemRestore [illegible] " & Drive & " [illegible]"
Else
   objTextFile.WriteLine " : [illegible]!!! SystemRestore [illegible] " & Drive & " [illegible]
[illegible]"
End If

case "Y"
If (obj.Enable(Drive)) = 0 Then
   objTextFile.WriteLine " : SystemRestore [illegible] " & Drive & " [illegible]"
```

```
Else
  objTextFile.WriteLine " : □□□□□□□□!!! SystemRestore □□□ □□□□□ " & Drive & " □□□□□□□□□□
□□□□□□□□"
End If
End  Select
End if

'□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□
'□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□

objTextFile.Close
end sub
```

Вот, в принципе, и все. Приведенный в листинге сценарий очень подробно изложен, поэтому больше он описан не будет. Для запуска данного сценария необходимо иметь права администратора на локальном компьютере, иначе вы не сможете получить доступ к WMI. Если немножко изменить этот сценарий, то можно будет подключиться не к локальному компьютеру, а к любому другому компьютеру сети. Для этого нужно строку

```
Set obj = GetObject("winmgmts:{impersonationLevel=impersonate}!root/
default:SystemRestore")
```

заменить строкой

```
Set obj = GetObject("winmgmts:{impersonationLevel=impersonate}!\\
«□□□ □□□□□□□□□□»\root\default:SystemRestore")
```

При этом вместо имени компьютера можно использовать точку. В этом случае вы опять будете подключаться к локальному компьютеру.

Напоследок несколько возможных команд запуска приведенного сценария:

- `cscript d:\sr.vbs` — полностью отключить возможность восстановления системы (если диск C: является системным);
- `cscript d:\sr.vbs c:\ Y` — полностью включить возможность восстановления системы (если диск C: является системным);
- `cscript d:\sr.vbs D:\ N` — отключить SystemRestore только на диске D:.

Создание контрольной точки восстановления

Продолжим рассмотрение темы восстановления системы и опишем код для автоматического создания точки восстановления. На этот раз будет приведена только часть кода, отвечающая за работу WMI, а не весь код сценария, то есть обращение к моникеру `winmgmts` и работа с классом. Вы самостоятельно можете модифицировать приведенный выше код для того, чтобы он мог, например, автоматически включать работу SystemRestore, если возможность восстановления системы отключена, а потом уже выполнять создание контрольной точки восстановления.

11.2.

```
Set obj = GetObject("winmgmts:{impersonationLevel=impersonate}!root/
default:SystemRestore")

'
'
'
'
'
'
'

ErrorSRCode = obj.CreateRestorePoint("      " & Time(), 0, 100)
```

Энумерация содержимого ветви реестра

Стандартные методы объекта, предназначенные для доступа к системному реестру (`WScript.CreateObject("Wscript.Shell")`), один из которых, позволяющий считывать значения параметров из реестра, был рассмотрен в примере сценария для включения/отключения восстановления системы на логических дисках компьютера, позволяют выполнить базовые операции с реестром.

Но данный объект имеет один очень большой недостаток — с его помощью нельзя перечислить все параметры, расположенные в определенной ветви реестра. Этот недостаток ограничивал возможности применения сценариев сервера сценариев Windows, поэтому просто нельзя не рассказать о новом свойстве инструментария Windows, которое выполняет именно эту операцию.

ПРИМЕЧАНИЕ

Это может понадобиться в том случае, если необходимо значение не конкретного параметра реестра, а всех параметров одной ветви. При этом точно не известно, какие именно параметры могут находиться в данной ветви.

За выполнение перечисления параметров, расположенных в ветви реестра, отвечает метод `EnumValues`, принадлежащий классу `StdRegProv`. Данный класс определяет методы для доступа к реестру Windows XP (более функциональный аналог стандартного объекта Windows, рассмотренного выше) и принадлежит к пространству имен `Root\Default`. Мы не будем создавать целый работоспособный сценарий для описания работы данного метода — лучше создадим отдельную процедуру, которая будет выполнять перечисление параметров указанной ветви реестра, а также проверим ее работу с помощью записи в файл журнала выводимых значений.

11.3.

```
set objFS = CreateObject("Scripting.FIleSystemObject")

'
'  – enum_log.txt  d:\.

set objTextFile = objFS.CreateTextFile("d:\enum_log.txt", 8, True)

'

Set obj = GetObject("winmgmts:{impersonationLevel=impersonate}!root/Default:StdRegProv")

'RootKey,
'
'Wmegh
'  : HKEY_CLASSES_ROOT (0x80000000),
'HKEY_CURRENT_USER (0x80000001), HKEY_LOCAL_MACHINE (0x80000002),
'HKEY_USERS (0x80000003), HKEY_CURRENT_CONFIG (0x80000005), HKEY_DYN_DATA
(0x80000006).
'HKEY_CURRENT_USER.

RootKey = &H80000001

'
'  ;
'  (  Control Panel\Desktop),
'
'
'RootKey,
'

call EnumV("Control Panel\Desktop", objTextFile, RootKey)
objTextFile.Close


Sub EnumV(Path, objTextFile, RootKey)


objTextFile.WriteLine " :::: : HKEY_CURRENT_USER\" & Path & vbCrLf

'
'HKEY_CURRENT_USER\Control Panel\Desktop.
```

EnumValues, [illegible]
'– [illegible] ([illegible] RootKey)
'– [illegible] ([illegible] RootKey,
' [illegible])
'– [illegible]
' [illegible] Names
'– [illegible]
' [illegible] Types

```
obj.EnumValues RootKey, Path, Names, Types
```

' [illegible]
' [illegible] Names, [illegible]
' [illegible] Types
' [illegible]
' [illegible] EnumValues [illegible]

```
if not IsNull(Names) and not IsNull(Types) Then
```

' [illegible]
' [illegible] types ([illegible] types, [illegible]
' [illegible])
' [illegible]

```
    for i = lbound(types) to ubound(types)
```

' [illegible]
' [illegible] types [illegible]
' [illegible] select case [illegible] EnumValues.
' [illegible]
' [illegible] types ([illegible]),
' [illegible]
' [illegible] Wmregh. [illegible]
'1 – [illegible];
'2 – [illegible];
'3 – [illegible] REG_BINARY-[illegible];
'4 – [illegible] REG_DWORD-[illegible];
'7 – [illegible] REG_MULTI_SZ-[illegible].

```
    select case types(i)
```

' [illegible]
' [illegible]

```
'
StdRegProv.
'GetStringValue –
'GetExpandedStringValue –
'GetBinaryValue –                                    REG_BINARY
'GetDWordValue –                                     REG_DWORD
'GetMultiStringValue –                               REG_MULTI_SZ
'
'
'                                    names,
'
'
```

```
 case 1
obj.GetStringValue RootKey, path, names(i), value
```

```
'
'                                  (                                  names),
'
'
'
```

```
If not isnull(names(i)) or not isnull(value) then
    objTextFile.WriteLine  names(i) & " = REG_SZ: " & value
end if
case 2
obj.GetExpandedStringValue RootKey, path, names(i), value
if not isnull(names(i)) or not isnull(value) then
    objTextFile.WriteLine names(i) & " = REG_EXPAND_SZ: " & value
end if
 case 3
obj.GetBinaryValue RootKey, path, names(i), value
for j = lbound(value) to ubound(value)
    value(j) = hex(cint(value(j)))
next
if not isnull(names(i)) or not isnull(value) then
    objTextFile.WriteLine names(i) &" = REG_BINARY : "& _
            join(value, ",")
end if
 case 4
```

```
            obj.GetDWordValue RootKey, path, names(i), value
            if not isnull(names(i)) or value then
                objTextFile.WriteLine names(i) & " = REG_DWORD : " & _
                        hex(value)
            end if
            case 7
            obj.GetMultiStringValue RootKey, path, names(i), value
            for j = lbound(value) to ubound(value)
                value(j) = value(j)
            next
            if not isnull(names(i)) or not isnull(value) then
                objTextFile.WriteLine names(i) &" = REG_MULTI_SZ : "& _
                        join(value, ",")
            end if
        end select
        next
    end if

    End Sub
```

При написании сценария был использован не только метод `EnumValues`, но и другие методы класса `StdRegProv`. Это было необходимо для занесения в текстовый файл значений параметров, найденных в данной ветви реестра. Но это не все методы, описанные в классе `StdRegProv`. И поскольку большую часть книги все-таки составляют описания параметров реестра, хотя бы вкратце рассмотрим другие методы данного класса. Класс `StdRegProv` содержит следующие методы.

- `CreateKey` — создает раздел в ветви реестра. Для его вызова необходимы следующие два параметра: идентификатор корневого раздела ветви реестра (аналог переменной `RootKey` приведенного сценария), а также остальной путь к ветви реестра, которую нужно создать (в том числе и сам создаваемый раздел реестра).

ПРИМЕЧАНИЕ

Если идентификатор корневого раздела не указан, то будет использоваться стандартный идентификатор &H80000002, говорящий о том, что ветвь находится в корневом разделе HKEY_LOCAL_MACHINE.

- `DeleteKey` — удаляет раздел в ветви реестра. Для его вызова также необходимы следующие два параметра: идентификатор корневого раздела ветви реестра (аналог переменной `RootKey` приведенного сценария), а также остальной путь к ветви реестра, которую нужно удалить (в том числе и сам удаляемый раздел реестра).

- `EnumKey` — в сценарии был рассмотрен метод для энумерации параметров указанной ветви реестра, этот же метод возвращает разделы, вложенные в указанную ветвь реестра. Для его работы необходимы следующие параметры: идентификатор корневого раздела ветви реестра (аналог переменной `RootKey` приведенного сценария), остальной путь к ветви реестра, разделы которой нужно перечислить, а также переменная, используемая для хранения массива найденных в указанной ветви разделов (например, аналог переменной `names` нашего сценария).
- `DeleteValue` — удаляет указанный параметр из ветви реестра. Для его работы необходимо указать следующие три параметра: идентификатор корневого раздела ветви реестра (аналог переменной `RootKey` приведенного сценария), остальной путь к ветви реестра, содержащей удаляемый параметр, а также название удаляемого параметра (если название не указано, то будет удалено значение параметра `(Ïî óìîë÷àíèþ)`).
- `SetDWORDValue` — создает или изменяет указанный параметр `DWORD`-типа в ветви реестра. Для его работы необходимо указать следующие четыре параметра: идентификатор корневого раздела ветви реестра (аналог переменной `RootKey` приведенного сценария), остальной путь к ветви реестра, содержащей изменяемый параметр, название изменяемого параметра (или создаваемого), а также новое значение, которое необходимо присвоить указанному параметру. Если название изменяемого параметра указано не будет, то нужно изменить значение параметра `(Ïî óìîë÷àíèþ)` данной ветви реестра.
- `CheckAccess` — определяет, разрешен ли пользователю доступ к указанной ветви реестра. Если метод выполнился успешно, то он возвратит значение `0`, иначе — любое другое значение. Для его работы необходимы следующие четыре параметра: идентификатор корневого раздела ветви реестра (аналог переменной `RootKey` приведенного сценария), остальной путь к ветви реестра, права на которую необходимо проверить, флаг для определения проверяемых прав, а также переменная, в которую будет помещен результат выполнения метода. Если значение этой переменной будет равно `true`, то пользователь обладает нужными правами на данную ветвь реестра, иначе возвращается значение `false`.

Флаг для определения проверяемых прав является битовой маской, значения которой определены в файле `Winnt.h`. Этот флаг может содержать следующие значения:

- `0X0001` — `KEY_QUERY_VALUE` (разрешено запрашивать значения из дочерних разделов вашей ветви реестра);
- `0X0002` — `KEY_SET_VALUE` (разрешено создание, удаление и установка значений параметров вашей ветви реестра);
- `0X0004` — `KEY_CREATE_SUB_KEY` (разрешено создание и удаление дочерних разделов вашей ветви реестра);
- `0X0008` — `KEY_ENUMERATE_SUB_KEYS` (разрешена энумерация дочерних разделов вашей ветви реестра);

- 0X0010 — KEY_NOTIFY (разрешен вывод подтверждения на запрос изменения параметров или дочерних разделов вашей ветви реестра);
- 0X0020 — KEY_CREATE_LINK (используется системой);
- 0x00010000 — DELETE (разрешено удаление);
- 0x00020000 — READ_CONTROL (разрешено управление чтением);
- 0X00040000 — WRITE_DAC (разрешен избирательный контроль над доступом к записи);
- 0X00080000 — WRITE_OWNER (разрешено изменение владельца ветви).

В классе StdRegProv также описаны методы SetBinaryValue, SetExpandedStringValue, SetMultiStringValue, SetStringValue, но мы их описывать не будем, так как их вызов аналогичен вызову описанного выше метода SetDWORDValue.

ПРИМЕЧАНИЕ

Подробнее о классе StdRegProv можно узнать из стандартного файла Windows XP regevent.mfl, расположенного в каталоге %systemroot%\SYSTEM32\wbem.

Создание, завершение и просмотр учетной записи процесса

К другим основным возможностям инструментария управления WMI относятся возможности работы с процессами, запущенными на удаленном или локальном компьютере. При этом инструментарий предоставляет возможности не только по запуску или завершению процесса, но и по определению учетной записи, от имени которой запущен определенный процесс. Для работы с процессами используется класс Win32_Process, принадлежащий к пространству имен root\cimv2. Рассмотрим маленький пример по созданию нового процесса с использованием метода Create.

11.4.

```
set obj = GetObject("winmgmts:" & "{impersonationLevel=impersonate}!\\.\root\
cimv2:Win32_Process")

obj.Create "C:\WINDOWS\regedit.exe"
```

Действительно, маленький пример. Теперь подробнее рассмотрим методы класса Win32_Process, направленные на работу с процессами.

- Create — как уже известно, данный метод используется для создания процесса. При этом после своего выполнения метод возвращает следующие значения (это не все):
  - 0 — успешное завершение;
  - 2 — пользователь не имеет доступа к запрашиваемым данным;

- `3` — у пользователя нет достаточных привилегий;
- `8` — неизвестная ошибка;
- `9` — пользователь не имеет необходимых привилегий;
- `21` — указан недопустимый параметр.

Несмотря на то, что при вызове данного метода был использован только один параметр, на самом деле для работы с ним нужны четыре параметра. Во-первых, это путь к файлу, который будет вызван (в данном случае применялся только этот параметр). Во-вторых, это путь к каталогу, который должен использоваться вызываемым процессом (если он не задан, то будет использоваться каталог, в котором находится запускаемый файл). В-третьих, это строка начальной конфигурации процесса (если она не задана, то будет применяться пустая строка (`\"\"`)). В-четвертых, это переменная, которой будет присвоен идентификатор созданного процесса.

- `Terminate` — прекращает процесс и все его дочерние процессы.
- `GetOwner` — возвращает учетную запись пользователя, от имени которого был запущен процесс, а также домен, к которому он принадлежит.
- `GetOwnerSid` — возвращает SID пользователя, от имени которого был запущен процесс.
- `SetPriority` — устанавливает приоритет выполнения определенного процесса.
- `AttachDebugger` — вызывает отладчик данного процесса.

ПРИМЕЧАНИЕ

За более детальной информацией обращайтесь к файлу cimwin32.mfl, расположенному в каталоге %systemroot%\SYSTEM32\wbem.

Выключение, перезагрузка компьютера, завершение сеанса пользователя

С помощью инструментария WMI можно выключить удаленный или локальный компьютер, перезагрузить его или завершить сеанс текущего пользователя. Рассмотрим пример завершения сеанса текущего пользователя на локальном компьютере (если у вас есть удаленный компьютер, то при подключении к пространству имен вместо точки необходимо указать его имя). Особенность данного примера состоит в другом способе подключения к классу — с помощью базы данных CIMOM.

11.5. root\cimv2

```
set objWMIService = GetObject("winmgmts:" & "{impersonationLevel =
impersonate}!\\.\root\cimv2")
```

```
' [illegible] SQL [illegible]

set colSoftware = objWMIService.ExecQuery("Select * from Win32_OperatingSystem")

' [illegible]
' [illegible] ( [illegible] ) [illegible]
' [illegible] ) – [illegible]
' [illegible] 0 [illegible]
' [illegible]
' [illegible]
' [illegible] SE_SHUTDOWN_NAME [illegible]
' [illegible] " [illegible] ",
" [illegible] "," [illegible] "," [illegible] "

for each objSoftware in colSoftware
   objSoftware.Win32Shutdown 0
next
```

Как правило, администраторы локального компьютера обладают правами на завершение сеанса. Другие же действия по умолчанию возможны от имени системы.

Кроме метода `Win32Shutdown`, класс `Win32_OperatingSystem` описывает еще три метода. Если вызывающий процесс обладает правом `SE_SHUTDOWN_NAME`, то выполняются следующие действия.

- `Reboot` — компьютер будет перезагружен. Метод не имеет параметров, но при успешном выполнении возвращает `0`.
- `Shutdown` — компьютер будет выключен. Метод не имеет параметров, но при успешном выполнении возвращает `0`.
- `SetDateTime` — будет установлено системное время компьютера. Данный метод для своей работы требует один параметр, определяющий новое время, которое нужно установить (имеет тип `DateTime`, имеющий довольно странный формат, например установка параметра равным `"00000000000000.000000:000"` обнуляет время).

Получение сведений об установленных на компьютере программах

Это еще одна интересная возможность WMI, которой можно воспользоваться. Другими словами, можно удаленно узнавать о том, установлена ли на компьютере какая-либо программа, когда она была установлена, а также просмотреть весь список установленных на компьютере программ. Для примера попробуем создать текстовый файл, содержащий сведения об установленных на компьютере программах, их описании, а также дате установки. В приведенном ниже коде будет использоваться

еще одна возможность WMI — получение сведений непосредственно из ее базы данных.

11.6.

```
'
'
'(  , Program_list.txt  e:\)

set objFSO = CreateObject("Scripting.FIleSystemObject")
set objTextFile = objFSO.CreateTextFile("e:\Program_list.txt", True)

'  root\cimv2

set objWMIService = GetObject("winmgmts:" & "{impersonationLevel = impersonate}!\
root\cimv2")

'  SQL-  \Win32_Product
'  ,  SQL-
'  ,
'

set colSoftware = objWMIService.ExecQuery("Select * from Win32_Product")

'
'  ,
'colSoftware

for each objSoftware in colSoftware
  objTextFile.WriteLine "  : " & objSoftware.Caption & "  (  :
" & objSoftware.Description & ")    :  " & objSoftware.InstallDate2
next
objTextFile.Close
```

После выполнения данного сценария в указанном каталоге появится текстовый файл с описанием установленных программ. При этом, кроме названия программы, ее описания и даты установки, можно получить и другую информацию об установленных программах. Например, можно воспользоваться такими столбцами:

- `objSoftware.IdenttifyiungNumber` — серийный номер программы;
- `objSoftware.InstallLocation` — каталог, в который установлена программа;
- `objSoftware.Name` — название программы, как правило, не отличается от `objSoftware.Caption`;

- `objSoftware.Vendor` — производитель программы;
- `objSoftware.Version` — версия программы.

Получение сведений о компьютере

С помощью базы данных CIMOM можно также определить конфигурацию операционной системы. Метод определения конфигурации подобен методу определения установленных на компьютере компонентов, поэтому будет рассмотрена лишь часть кода, с помощью которой выполняется обращение к базе данных.

11.7.

```
set objWMIService = GetObject("winmgmts:" & "{impersonationLevel = impersonate}!\
root\cimv2")
set colOS = objWMIService.ExecQuery("Select * from Win32_OperatingSystem")
for each objOperatingSystem in colOS
objTextFile.WriteLine objOperatingSystem.Name
next
```

Именно так можно подключиться к базе данных сведений об операционной системе и записать их в файл. Но к каким столбцам таблицы можно получить доступ (кроме используемого в листинге столбца `Name`, который содержит имя операционной системы и путь к папке Windows)? Ниже перечислены некоторые из таких столбцов (в контексте инструментария управления WMI они называются свойствами).

- `BootDevice` — указывает имя тома, из которого загружается операционная система.
- `CodeSet` — указывает кодовую страницу, используемую операционной системой (как написано в MFL-файле, кодовая страница содержит таблицу символов, применяемую операционной системой для преобразования строк на различных языках).
- `DataExecutionPrevention_Available` — если данное свойство возвращает значение `true`, то оборудование, установленное на компьютере, поддерживает технологию предотвращения выполнения данных Windows (DEP). Более подробные сведения об этой технологии можно найти в файле `cimwin32.mfl`.
- `LargeSystemCache` — указывает, для каких операций выполняется оптимизация памяти. Если свойство возвращает значение `0`, то оптимизация памяти выполняется для повышения производительности работающих программ. Если же возвращается значение `1`, то оптимизация памяти выполняется для повышения производительности работы операционной системы.

ПРИМЕЧАНИЕ

Свойство использует для определения метода оптимизации уже рассмотренный ранее параметр реестра LargeSystemCache, расположенный в ветви HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Session Manager\Memory Management. Если его значение равно 1, то возвращается значение 1.

- `Version` — содержит номер версии операционной системы Windows.
- `ServicePackMajorVersion` — основная версия установленного пакета обновлений.
- `ServicePackMinorVersion` — дополнительная версия установленного пакета обновлений.
- `Manufacturer` — имя производителя операционной системы.
- `WindowsDirectory` — каталог Windows.
- `Locale` — код локализации (419 для русской версии, 409 для английской).
- `FreePhysicalMemory` — свободный объем жесткого диска.
- `FreeVirtualMemory` — свободный объем виртуальной памяти.
- `TotalVirtualMemorySize` — общий объем виртуальной памяти.

Здесь были рассмотрены лишь 13 свойств класса `Win32_OperatingSystem`, на самом же деле данный класс имеет 35 свойств. Не имеет смысла описывать остальные свойства, так как все они отлично описаны в стандартном файле `cimwin32.mfl`, расположенном в каталоге `%systemroot%\SYSTEM32\wbem`, а автор не брал на себя задачу создания книжного аналога данного файла. Поэтому за дополнительной информацией обращайтесь к файлу `cimwin32.mfl` (просто поищите в нем строку `Win32_OperatingSystem`).

Кроме `Win32_OperatingSystem`, для описания компьютера можно использовать следующие классы.

- `Win32_ComputerSystem` — содержит следующие свойства, описывающие работающий компьютер.
  - `AutomaticResetBootOption` — если данное свойство возвращает значение `false`, то при возникновении аварийной остановки отображается «синий экран смерти» (BSOD). Если же значение равно `true`, то компьютер автоматически перезагружается.
  - `BootupState` — определяет способ загрузки операционной системы. Например, если данное свойство возвращает значение `Normal Boot`, то операционная система была загружена в обычном режиме. Возможны следующие значения: `Normal boot`, `Fail-safe boot`, `Fail-safe with network boot`.
  - `Name` — имя компьютера.
  - `NumberOfProcessors` — возвращает количество процессоров, установленных на данном компьютере.
  - `Manufacturer` — имя компании, собиравшей компьютер.
  - `Model` — модель BIOS компьютера (поддерживает ACPI или нет).
  - `CurrentTimeZone` — идентификатор текущей зоны часового пояса.
  - `TotalPhysicalMemory` — общий объем физической памяти.

ПРИМЕЧАНИЕ

Класс содержит 39 свойств, поэтому за описанием других свойств обращайтесь к файлу cimwin32.mfl, расположенному в каталоге %systemroot%\SYSTEM32\wbem. Просто поищите в нем строку Win32_ComputerSystem.

- `Win32_Processor` — определяет один экземпляр процессора (для многопроцессорных систем существует несколько экземпляров данного класса) и содержит следующие свойства:
  - `Description` — описание процессора, установленного на компьютере;
  - `Architecture` — тип процессора, установленного на компьютере;
  - `CurrentVoltage` — возвращает текущее напряжение, используемое процессором (определяется первыми 6 байтами, умноженными на 10);
  - `L2CacheSize` — возвращает размер кэша второго уровня для данного процессора;
  - `LoadPercentage` — возвращает среднюю величину загруженности процессора в течение одной секунды.

ПРИМЕЧАНИЕ

Класс содержит 16 свойств, поэтому за описанием других свойств обращайтесь к файлу cimwin32.mfl, расположенному в каталоге %systemroot%\SYSTEM32\wbem. Просто поищите в нем строку Win32_Processor.

- `Win32_BIOS` — указывает атрибуты служб ввода/вывода, установленных на компьютере и содержит следующие свойства:
  - `Version` — описание версии BIOS материнской платы;
  - `CurrentLanguage` — возвращает имя текущего языка BIOS.

ПРИМЕЧАНИЕ

Класс содержит 11 свойств, поэтому за описанием других свойств обращайтесь к файлу cimwin32.mfl, расположенному в каталоге %systemroot%\SYSTEM32\wbem.

- `Win32_OSRecoveryConfiguration` — определяет установленные настройки выполнения дампа памяти при аварийной остановке системы и содержит следующие свойства.
  - `DebugFilePath` — возвращает путь к файлу дампа памяти, который будет создаваться при возникновении аварийной остановки.
  - `MiniDumpDirectory` — возвращает каталог, который используется для хранения малых дампов памяти.

- `WriteToSystemLog` — указывает, будет ли выполняться запись в системный журнал событий при возникновении аварийной остановки. Если возвращает значение `false`, то не будет.

ПРИМЕЧАНИЕ

Класс содержит 11 свойств, поэтому за описанием других свойств обращайтесь к файлу cimwin32.mfl, расположенному в каталоге %systemroot%\SYSTEM32\wbem.

- `Win32_Process` — указывает запущенные в данный момент на удаленном или локальном компьютере процессы и содержит следующие свойства, возвращающие:
  - `ExecutablePath` — пути к исполняемым файлам процессов, запущенных в данный момент;
  - `MaximumWorkingSetSize` — максимальный набор страниц памяти, доступных процессам;
  - `PageFaults` — количество ошибок страниц, которые были допущены в течение всего времени работы процесса;
  - `PageFileUsage` — объем файла подкачки, который используется процессом в данный момент;
  - `ProcessId` — идентификатор процесса;
  - `QuotaPagedPoolUsage` — размер используемой процессом в данный момент части выгружаемого пула;
  - `CommandLine` — командную строку, которая использовалась для запуска данного процесса.

ПРИМЕЧАНИЕ

Класс содержит 30 свойств, поэтому за описанием других свойств обращайтесь к файлу cimwin32.mfl, расположенному в каталоге %systemroot%\SYSTEM32\wbem.

- `Win32_StartupCommand` — определяет файлы, запускаемые при входе пользователя в систему. При этом возвращаются не только исполняемые файлы, но и файлы других типов (независимо от расширения файла, если исполняемая программа обращается к данному файлу, то класс считает, что файл запускается при входе пользователя в систему). Например, на компьютере автора данный класс вернул около 4582 файлов, которые запускаются при входе пользователя в систему. Класс содержит следующие свойства.
  - `Command` — возвращает командную строку, с помощью которой запускаются исполняемые файлы, или имя файла, который запускается исполняемым.
  - `User` — имя учетной записи пользователя, при входе которого запускается данный файл (так описание свойства определено в файле `cimwin32.mfl`,

хотя возвращаемое значение больше похоже на имя учетной записи, с правами которой данный файл запускается).

- `Name` — название запускаемого файла (без расширения).
- `Location` — значение `Startup`, говорящее о том, что данный файл запускается с помощью папки Автозагрузка, или ветвь реестра, из которой выполняется запуск данного файла.

- `Win32_NTEventlogFile` — определяет параметры настройки системных журналов (рассмотренная оснастка Просмотр событий) и содержит следующие свойства:
  - `LogFileName` — возвращает имя системного журнала;
  - `MaxFileSize` — определяет предельный размер файла системного журнала;
  - `NumberOfRecords` — возвращает количество записей, хранящихся в данный момент в системных журналах;
  - `OverwriteOutDated` — количество дней, в течение которого запись может храниться в системном журнале.
- `Win32_AccountSID` — определяет учетные записи и группы, созданные на данном компьютере. Класс содержит следующие два свойства.
  - `Element` — перечисляет все доступные на компьютере группы и учетные записи. При этом разница между данными объектами отображается в виде класса, к которому они принадлежат (формат вывода в текстовый файл таков: пространство имен WMI и класс, к которому принадлежит данная учетная запись или группа, а также название данной учетной записи или группы, например, `\\ME\root\cimv2:Win32_Group.Domain="ME",Name="Àäìèíèñòðàòîð"`).
  - `Setting` — перечисляет SID учетных записей и групп, созданных на данном компьютере.

Это далеко не все возможные классы и их свойства. Автор не брал на себя ответственность по полному их описанию. Тем не менее уже по описанным свойствам можно понять, что инструментарий управления WMI позволяет получить описание если не всех параметров работы компьютера и установленных на нем приложений, то большинства из них. А если учитывать, что WMI постоянно развивается, то скоро от ее глаз не скроется ни одна мелочь в работе компьютера.

Другие классы и функции пространств имен WMI

WMI является неисчерпаемой темой для обсуждения, так как содержит просто огромное количество классов, не говоря уже о количестве функций, реализованных в этих классах. Для рассмотрения всех функций WMI (не говоря уже об объектах сервера сценариев Windows для доступа к файловой системе и реестру и их методах) необходимо писать отдельную книгу страниц где-то на 1000. Поэтому в контексте данной книги мы закончим обсуждение базы данных CIMOM и пространств имен WMI. Если же приведенных выше сценариев для вас мало, то в Интернете можно найти множество уже готовых сценариев работы с WMI. Кроме того, список всех классов, их функций, а также краткое описание работы этих функций

можно найти в самой Windows XP. Краткие сведения обо всех функциях пространств имен WMI содержатся в файлах с расширениями MOF и MFL (в файлах с расширением MOF хранится описание функций на английском языке, а в файлах с расширением MFL — на русском), расположенных в каталоге `%systemroot%\system32\wbem`. Например, среди данных файлов можно найти файл `sr.mof`. В нем содержится описание всех классов, которые предназначены для работы с SystemRestore. Например, вот небольшая вырезка из файла, описывающая рассмотренный выше класс SystemRestore.

11.8. SystemRestore sr.mof

```
[Dynamic, Provider ("SystemRestoreProv")]
class SystemRestore
{
  [read, write]
   String Description;
  [read, write]
   uint32 RestorePointType;
  [read, write]
   uint32 EventType;
  [read, write, key]
   uint32 SequenceNumber;
  [read, write]
   String CreationTime;

  [Implemented, static, Description(
   "The CreateRestorePoint method creates a restore point."
  "It returns a COM error code.")]
   uint32 CreateRestorePoint([In] String Description, [In] uint32 RestorePointType, [In] uint32
EventType );

  [Implemented, static, Description(
   "The Enable method enables SR on a drive."
  "It returns a COM error code.")]
   uint32 Enable([In] String Drive, [In] Boolean WaitTillEnabled);

  [Implemented, static, Description(
   "The Disable method disables SR on a drive."
  "It returns a COM error code.")]
   uint32 Disable([In] String Drive);

   [Implemented, static, Description(
   "The Restore method restores the system to a specified restore point."
```

```
    "It returns a COM error code.")]
    uint32 Restore([In] uint32 SequenceNumber);

    [Implemented, static, Description(
    "Returns the status (0=fail, 1=success, 2=interrupted) of the last restore.")]
    uint32 GetLastRestoreStatus();

};
```

Как заметно из данной вырезки, с помощью класса SystemRestore можно определить, была ли успешной предыдущая попытка восстановления системы из контрольной точки, а также можно автоматически восстановить систему на основе указанной контрольной точки. К сожалению, не существует файла `sr.mfl`, хранящего описание на русском языке. Зато в каталоге `%systemroot%\system32\wbem` есть файл `cimwin32.mfl`, содержащий описание всех функций и классов пространства имен `Root\CIMV2` на русском языке. В каталоге `%systemroot%\system32\wbem` также находятся следующие интересные файлы.

- `regevent.mfl` — описывает работу с реестром Windows XP.
- `licwmi.mfl` — хранит свойства и методы для работы с функцией активизации Windows XP. В частности, метод для автоматической активизации операционной системы через Интернет.
- `Msi.mfl` — описывает свойства и методы для работы с пакетами установки Windows. Некоторые свойства из этого файла (класс `Win32_Product`) были рассмотрены ранее. Методы же, которые он поддерживает, позволяют удаленно установить пакет установщика Windows с правами системы.
- `ntevt.mfl` — хранит свойства и методы для работы с системными файлами журналов. В частности, методы очистки системных журналов или их копирования.
- `rsop.mfl` — описывает свойства и методы для работы с результирующей политикой (RSoP).
- `secrcw32.mfl` — хранит свойства и методы для работы с учетными записями компьютера, а также списками ACL и другими объектами безопасности компьютера.
- `smtpcons.mfl` — описывает свойства для работы с почтовыми сообщениями smtp-сервера.

Стандартные сценарии сервера сценариев

Напоследок хотелось бы описать стандартные сценарии сервера сценариев Windows XP (поставляются вместе с Windows XP Professional). Все они расположены в каталоге `%systemroot%\system32`.

- `Eventquery.vbs` — предназначен для работы с системными журналами событий (оснастка Просмотр событий) и при запуске без параметров, например `cscript.exe C:\WINDOWS\SYSTEM32\eventquery.vbs`, перечисляет все

события, записанные в локальном журнале Система. Чтобы просмотреть возможные параметры работы данного сценария, необходимо ввести команду `cscript.exe C:\WINDOWS\SYSTEM32\eventquery.vbs /?`. Честно сказать, количество параметров впечатляет. Можно подключаться к удаленному компьютеру, для этого указать учетную запись и пароль, настроить фильтр выводимых записей событий, способ вывода событий и т. д. Сценарий хорошо описан, да к тому же еще и на русском языке, а также содержит набор примеров, поэтому его использование сложности не представляет.

Его также хорошо использовать для изучения работы инструментария управления WMI, так как данный сценарий написан на его основе, хотя размер сценария в 2080 строк не вдохновляет на изучение.

- `Pagefileconfig.vbs` — используется для работы с файлом подкачки как локального, так и удаленного компьютера. При вызове без параметров отображает сведения о локальном файле подкачки: диск, на котором он расположен, путь и его имя, исходный размер файла подкачки, максимальный размер, а также размер файла подкачки на данный момент. Определяет количество свободного места на логическом диске.

  Можно также выполнить команду `cscript.exe C:\WINDOWS\SYSTEM32\pagefileconfig.vbs /?` для вывода описания возможных параметров данного сценария. Эти параметры объединены в группы, описание которых можно вызвать с помощью следующих команд.

  - `cscript C:\WINDOWS\SYSTEM32\pagefileconfig.vbs /change /?` — описывает функции для изменения размера файла подкачки на локальном или удаленном компьютере.
  - `cscript C:\WINDOWS\SYSTEM32\pagefileconfig.vbs /create /?` — создать или добавить новый файл подкачки на логическом диске локального или удаленного компьютера.
  - `cscript C:\WINDOWS\SYSTEM32\pagefileconfig.vbs /delete /?` — удалить файл подкачки на логическом диске локального или удаленного компьютера.
  - `cscript C:\WINDOWS\SYSTEM32\pagefileconfig.vbs /query /?` — отобразить текущие параметры файла подкачки на удаленном или локальном компьютере (если на локальном, то данная функция эквивалентна вызову сценария без параметров).

  Сценарий написан при помощи инструментария управления WMI, но его размер в 3302 строчки также не способствует изучению.

- `Prncnfg.vbs` — предназначен для выполнения конфигурации принтера, установленного на локальном компьютере: изменения его имени, порта, приоритета и многих других параметров.
- `Prndrvr.vbs` — используется для выполнения настройки драйверов принтера, установленного на локальном компьютере: удаление, добавление, перечисление драйверов.

- `Prnjobs.vbs` — предназначен для работы с заданиями принтера: просмотр всех заданий, приостановка и продолжение выполнения задания, а также отмена выполнения задания.
- `Prnmngr.vbs` — используется для подключения и отключения принтеров (установленных на удаленном компьютере).
- `Prnport.vbs` — предназначен для подключения и отключения TCP-порта принтера.
- `Prnqctl.vbs` — позволяет выполнить пробную печать на данном принтере, а также приостановить работу принтера.

Просмотр событий

С помощью оснастки Просмотр событий реализуется возможность просмотра трех системных журналов Windows XP, каждый их которых содержит сведения о том или ином компоненте компьютера: о приложениях, установленных на нем, о работе системных компонентов и о работе компонентов безопасности компьютера. Оснастка Просмотр событий входит в стандартную консоль Windows XP `eventvwr.msc` и имеет GUID-номер `{975797FC-4E2A-11D0-B702-00C04FD8DBF7}`.

Запуск консоли

После ввода в командной строке команды `eventvwr.msc` консоль управления Microsoft просматривает две дополнительные ветви реестра (естественно, что также она просматривает ветви реестра, относящиеся к настройке самой оснастки Просмотр событий).

Во-первых, просматривается содержимое ветви реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Eventlog`, хранящей настройки службы Журнал событий. Данная ветвь системного реестра включает в себя три раздела: `Application`, `Security`, `System`. Все три раздела просматриваются консолью управления Microsoft на существование параметра `File` расширенного строкового типа. Именно этот параметр и содержит путь к файлу журнала, хранящему все сведения, отображаемые в одном из журналов оснастки Просмотр событий. Соответственно параметр `File` раздела `Application` хранит путь к журналу приложения, параметр раздела `Security` — путь к журналу безопасности, а параметр раздела `System` — путь к журналу системы. Остальные параметры этой ветви реестра будут рассмотрены далее.

Во-вторых, просматривается содержимое ветви реестра `HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Microsoft\EventViewer`, параметры которой будут рассмотрены далее в этом разделе.

Если запуск оснастки Просмотр событий не запрещен, то перед вами отобразится окно, подобное приведенному на рис. 11.6.

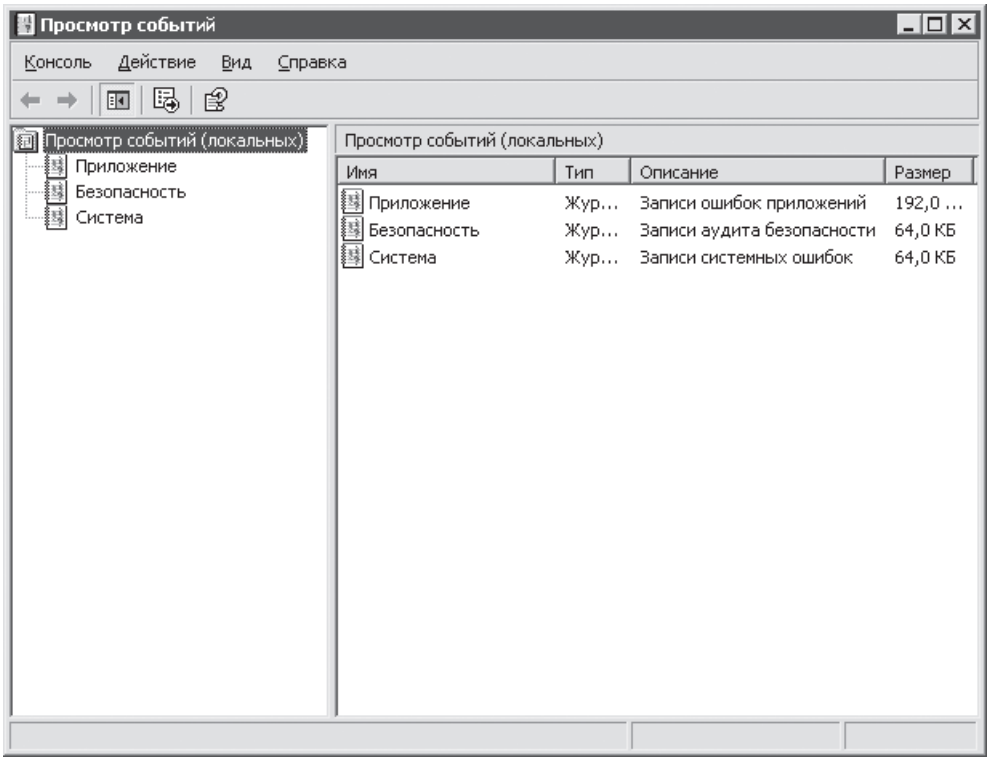


Рис. 11.6. Окно консоли Просмотр событий

ПРИМЕЧАНИЕ

С помощью добавления данной оснастки в консоль управления Microsoft можно просмотреть журналы событий другого компьютера. Это можно также сделать с помощью команды Подключиться к другому компьютеру меню Действие консоли Просмотр событий.

Дерево консоли Просмотр событий включает в себя три элемента — Приложение, Безопасность и Система. Выбор одного из этих элементов приведет к отображению на правой панели консоли содержимого соответствующего журнала, определенного в параметре `File` строкового типа рассмотренной выше ветви реестра. С помощью контекстного меню журналов дерева консоли можно также выполнить такие операции, как сохранение сведений из журнала (команда Сохранить файл журнала как), очистка журнала (команда Стереть все события), а также загрузка сведений из сохраненного ранее журнала (команда Открыть файл журнала).

Отдельно стоит сказать о команде контекстного меню Свойства. После выбора данной команды перед вами отобразится диалог, в котором можно изменить некоторые параметры журнала: имя, максимальный размер, а также определить интер-

вал времени, по истечении которого журнал будет очищаться. С помощью данного диалога можно очистить журнал, а также узнать текущий размер журнала, дату его создания и изменения.

Правая панель консоли состоит из нескольких столбцов. Наиболее интересны из них следующие.

Тип. Столбец определяет тип, к которому относится соответствующая запись журнала. Существует три типа записей: Уведомление, Предупреждение и Ошибка. Записи типа Уведомление, как правило, несут информативный характер и описывают какое-либо событие, которое должно было произойти (то есть системе известно это событие). Записи Предупреждение, как правило, описывают событие, которое не произошло по непонятным причинам. Записи типа Ошибка, как правило, описывают события, после выполнения которых дальнейшая работа какой-либо функции программы или системы невозможна.

По умолчанию отображаются записи журнала всех типов, хотя существует возможность указания вывода лишь записей отдельных типов. Для этого необходимо в контекстном меню определенного журнала (в дереве консоли) выбрать команду Вид ▸ Фильтр. После этого отобразится диалоговое окно, в котором можно настроить фильтр выводимых значений не только по полю Тип, но и по всем другим полям, описанным ниже.

Дата и Время. Эти два столбца определяют дату и время, когда произошло соответствующее событие.

Источник. Столбец определяет название программы, службы или компонента, в работе которого произошло описываемое событие. Названия всех возможных источников хранятся в реестре, и далее в этом разделе они еще будут рассмотрены.

Правая панель консоли определяет лишь заголовок сообщения журнала. Можно также прочитать описание соответствующей записи журнала. Для этого достаточно в контекстном меню необходимой записи выбрать команду Свойства. Например, на рис. 11.7 отображено окно описания для записи журнала Приложение.

В открывшемся диалоге отображается большая часть полей правой панели консоли, а также область описания записи и поле части содержимого памяти, ошибка или событие в которой породило данную запись (например, запись на рис. 11.7 вызвал ActiveX-объект с GUID-номером, отображенным в поле части содержимого памяти). Стоит также сказать о поле описания записи. Для записей из журнала Приложение данное поле иногда (если приложение, породившее данную запись журнала, написано корпорацией Microsoft) может хранить адрес, по которому можно найти дополнительную информацию о данной ошибке (например, запись на рис. 11.7). Адрес будет передаваться программе `event.asp`, которая и будет открывать адрес (или программу). По умолчанию используется адрес сайта Microsoft, хотя, как можно видеть на рис. 11.7, этот адрес можно переопределить.

Для этого и применяется ветвь системного реестра `HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Microsoft\EventViewer`, которая содержит следующие параметры.

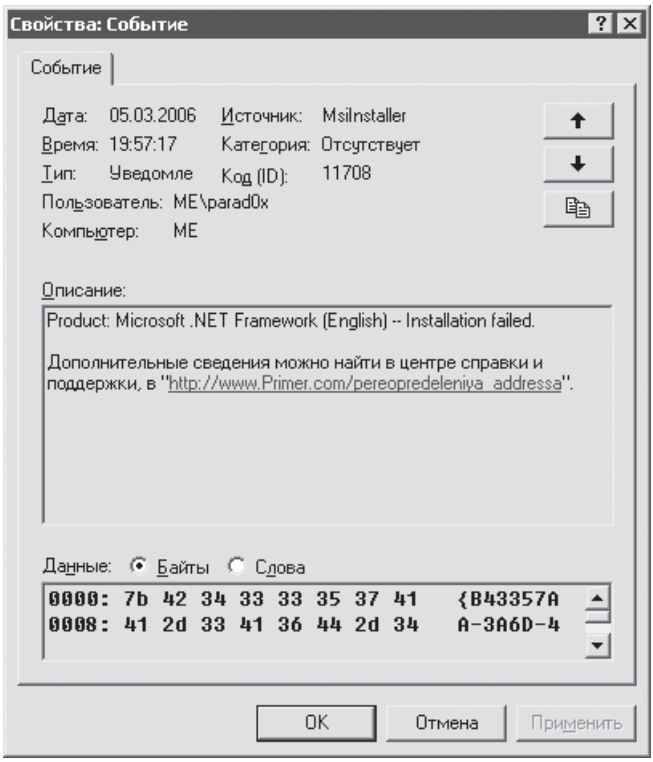


Рис. 11.7. Диалог описания записи журнала

ПРИМЕЧАНИЕ

Данные параметры используются, начиная с Windows XP Service Pack 2.

- `MicrosoftRedirectionURL` — этот параметр строкового типа определяет адрес сайта, который будет отображаться в поле описания. Адрес может не совпадать с реальным адресом (или командой), на который вы перейдете после щелчка на ссылке. По умолчанию используется адрес `http://go.microsoft.com/fwlink/events.asp`.
- `MicrosoftRedirectionProgram` — параметр строкового типа, указывает реальный адрес сайта (или путь к программе), страница которого будет открываться при щелчке на данной ссылке. Значение данного параметра будет передаваться программе `event.asp`. По умолчанию в значении хранится путь к программе `%SystemRoot%\PCHealth\HelpCtr\Binaries\HelpCtr.exe`.

- MicrosoftRedirectionProgramCommandLineParameters — этот параметр строкового типа определяет командную строку, которая будет передаваться программе event.asp. По умолчанию используется значение -url hcp://services/centers/support?topic=%s.
- MicrosoftEventVwrDisableLinks — если значение данного параметра DWORD-типа будет равно 1, то ссылки дополнительной информации в поле описания отображаться не будут.

Напоследок хотелось бы напомнить о возможности ведения аудита доступа к тем или иным объектам файловой системы или реестра Windows XP. Все события аудита доступа будут записываться в журнал безопасности системы.

Чтобы определить аудит изменения, удаления, чтения и т. д. параметров ветви реестра, необходимо выбрать в контекстном меню данной ветви команду Разрешения. После этого перед вами отобразится диалог Разрешения, в котором нужно нажать кнопку Дополнительно. Затем откроется окно Дополнительные параметры безопасности, в котором нужно перейти на вкладку Аудит и нажать на ней кнопку Добавить. После этого система попросит указать учетную запись пользователя, а затем выведет диалог для выбора событий, возникновение которых приведет к записи в системный журнал безопасности информации о данных событиях.

Чтобы задать аудит доступа для папки или файла, необходимо в его диалоге Свойства перейти на вкладку Безопасность (данная вкладка доступна только на логических дисках, отформатированных под NTFS) и нажать на ней кнопку Дополнительно. После этого отобразится диалоговое окно Дополнительные параметры безопасности, работа с которым была описана выше.

Ветви реестра, используемые оснасткой

Теперь вкратце рассмотрим структуру ветви реестра HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Eventlog, содержащей настройки службы Журнал событий, используемой в работе оснастки Просмотр событий. Как уже говорилось, данная ветвь реестра включает в себя три раздела (Application, Security и System). Эти разделы могут хранить следующие параметры.

- File — путь к файлу журнала. Этот параметр уже описывался выше.
- MaxSize — параметр DWORD-типа, определяет максимальный размер файла данного журнала.
- RestrictGuestAccess — этот параметр DWORD-типа определяет, разрешено ли пользователю учетной записи Гость просматривать содержимое журналов. По умолчанию значение для всех журналов системы равно 1, то есть гостю доступ к журналам запрещен.
- Retention — параметр DWORD-типа, определяет интервал времени в часах, по истечении которого система будет очищать данный журнал.
- Sources — этот параметр REG_MULTI_SZ-типа хранит имена всех источников записей, которые могут присутствовать в данном журнале. Если быть точным,

то значение этого параметра определяет названия разделов, вложенных в данный раздел. Параметры, содержащиеся в этих разделах, осуществляют поддержку записей от соответствующих источников. Если удалить источник из значения параметра `Sources`, то записи о нем не будут помещаться в данное приложение.

Кроме параметров, разделы журналов включают в себя другие разделы, описывающие программы, службы или компоненты операционной системы, которые могут быть источниками записей для данного журнала. Эти разделы содержат параметры, описывающие библиотеку, предназначенную для работы с соответствующим источником записей (параметры строкового типа `EventMessageFile` и `ParameterMessageFile`).

Редактор объекта групповой политики

Оснастка поставляется только с операционной системой Windows XP, входит в стандартную консоль `gpedit.msc` и имеет GUID-номер `{8FC0B734-A0E1-11D1-A7D3-0000F87571E3}`. Доступ к ней имеют только администраторы. С помощью данной оснастки можно запретить те или иные компоненты операционной системы Windows XP. Принцип ее работы довольно интересен, так как все ограничения на компоненты Windows XP заносятся не только в реестр, но и в два специальных файла. Это файлы с именем `Registry.pol`, расположенные в каталогах `%systemroot%\system32\GroupPolicy\Machine` и `%systemroot%\system32\GroupPolicy\User` (запись в данные каталоги разрешена только администраторам). Через определенные промежутки времени настройки из этих файлов заносятся в реестр (можно также воспользоваться командой `gpupdate.exe`, после выполнения которой настройки из файлов будут вручную внесены в реестр). Другими словами, если вы настроите ограничения с помощью групповых политик, а потом удалите из реестра параметры ограничений, то через некоторое время они опять будут записаны в реестр. Поэтому для снятия ограничений нужно либо пользоваться оснасткой Редактор объекта групповой политики, либо удалять также и файлы `Registry.pol`.

ПРИМЕЧАНИЕ

Если компьютер находится в домене, то на него действуют не только локальные, но и групповые политики на уровне домена и организационной единицы. При этом политики на уровне домена будут замещать собой политики локального компьютера (если изменяемые параметры реестра в этих политиках будут совпадать), а групповые политики на уровне организационной единицы будут замещать собой политики домена.

В книге будет рассказано лишь о локальном применении групповых политик.

Запуск консоли gpedit.msc

После запуска консоли `gpedit.msc` перед вами отобразится окно, изображенное на рис. 11.8.

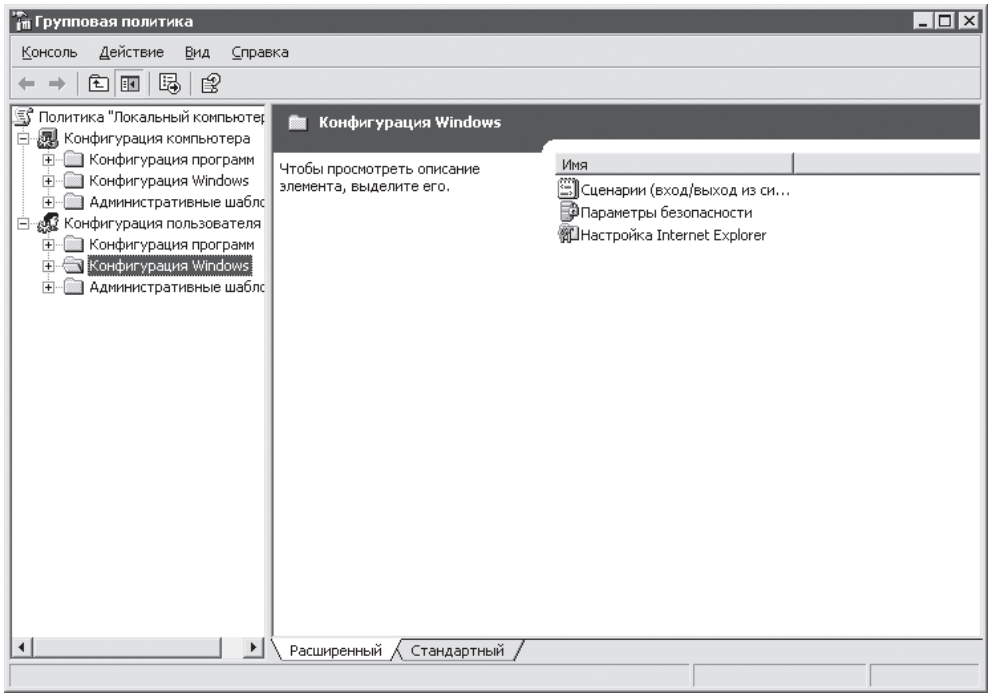


Рис. 11.8. Окно консоли Групповая политика

ПРИМЕЧАНИЕ

С помощью добавления данной оснастки в консоль управления Microsoft можно просмотреть журналы событий другого компьютера.

Дерево консоли этого окна включает в себя два элемента — Конфигурация компьютера и Конфигурация пользователя. Как правило, эти элементы хранят одни и те же вложенные элементы: Конфигурация программ, Конфигурация Windows и Административные шаблоны. При этом если вы воспользуетесь содержимым элемента Конфигурация компьютера, то все изменения, вносимые вами, будут заноситься в корневой раздел реестра HKEY_LOCAL_MACHINE. Если же вы воспользуетесь содержимым элемента Конфигурация пользователя, то все изменения будут заноситься в корневой раздел HKEY_CURRENT_USER.

Работа с консолью

Вкратце рассмотрим содержимое элементов Конфигурация компьютера и Конфигурация пользователя.

Конфигурация программ

Элемент предназначен для хранения назначенных или опубликованных администратором программ (технология Software Installation). Если программа назначена,

то всегда при запуске компьютера ее ярлык будет создаваться на Рабочем столе, и если она понадобится пользователю, то ему будет достаточно запустить этот ярлык или открыть файл с расширением, ассоциированным с программой, после чего начнется ее установка. Если же программа опубликована, то ссылка на нее будет помещена на вкладку Установка программ диалога Установка и удаление программ. Именно с помощью этой вкладки пользователь сможет установить необходимую программу.

Технология Software Installation применяется только в том случае, если в сети развернута Active Directory и данный компьютер входит в домен. Иначе элемент Конфигурация программ всегда будет пуст. Тем не менее, ради интереса, существует возможность регистрации этой оснастки как изолированной. Для этого достаточно создать раздел `StandAlone` в ветвях реестра `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\MMC\SnapIns\{942A8E4F-A261-11D1-A760-00C04FB9603F}` и `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\MMC\SnapIns\{BACF5C8A-A3C7-11D1-A760-00C04FB9603F}`. После этого в консоль управления Microsoft (`mmc.exe`) можно будет добавить оснастки Установка программ (пользователи) и Установка программ (компьютеры). В контекстном меню добавленных оснасток присутствует команда Создать ▸ Пакет, после ее выбора вам предложат указать файл с расширением MSI, который будет назначаться или опубликовываться. После указания данного файла консоль управления Microsoft попытается получить к нему доступ по сети, а затем выведет диалог для выбора способа развертывания программы. И в самом конце консоль управления Microsoft попытается обратиться к Active Directory для развертывания программы.

Конфигурация Windows

Элемент содержит два вложенных раздела — Сценарии и Параметры безопасности.

Раздел Сценарии применяется для назначения программ или сценариев, которые будут автоматически запускаться:

- при загрузке (Автозагрузка) и выключении компьютера (Завершение работы), в этом случае сценарии описываются в элементе Конфигурация компьютера и запускаются с правами системы;
- при входе пользователя в систему (Вход в систему) и выходе из нее (Выход из системы), в этом случае сценарии описываются в элементе Конфигурация пользователя и запускаются от имени учетной записи зарегистрированного в системе пользователя.

Для примера попробуем назначить программу для запуска при выходе данного пользователя из системы. Для этого необходимо в контекстном меню раздела Выход из системы (Конфигурация пользователя ▸ Конфигурация Windows ▸ Сценарии) выбрать команду Свойства. После этого перед вами отобразится диалоговое окно, в котором можно будет добавить на исполнение новую программу (кнопка Добавить) либо удалить уже добавленную программу (кнопка Удалить). При добавлении программы необходимо будет указать имя программы, а также параметры ее запуска (если они необходимы).

Если вы назначаете на выполнение сценарий, расположенный на локальном компьютере, то желательно, чтобы он находился в одном из следующих каталогов. Сценарий должен исполняться:

- `%systemroot%\System32\GroupPolicy\User\Scripts\Logon` — при входе пользователя в систему;
- `%systemroot%\System32\GroupPolicy\User\Scripts\Logoff` — при выходе пользователя из системы;
- `%systemroot%\System32\GroupPolicy\Machine\Scripts\Shutdown` — при выключении компьютера;
- `%systemroot%\System32\GroupPolicy\Machine\Scripts\Startup` — при включении компьютера.

Но что же происходит при назначении автоматического запуска программ? Как и при назначении групповых политик, при назначении программы редактируется как содержимое реестра, так и содержимое специального файла. Изменяемое содержимое реестра довольно сложно как в понимании, так и при ручном создании назначения файла, поэтому рассмотрено оно не будет. А вот синтаксис специальных файлов довольно прост. Существует два файла для автозагрузки. Оба они называются `scripts.ini` (это скрытые файлы), но первый из них расположен в каталоге `%systemroot%\System32\GroupPolicy\User\Scripts`, а второй в каталоге `%systemroot%\System32\GroupPolicy\Machine\Scripts`. Как можно догадаться, первый файл предназначен для хранения программ, запускаемых при входе пользователя в систему и выходе из нее, а второй файл хранит программы, загружаемые при включении компьютера и завершении его работы. Для примера рассмотрим возможное содержимое файла `scripts.ini` из каталога `%systemroot%\System32\GroupPolicy\User\Scripts`.

11.9. scriptsini

```
[Startup]
0CmdLine=c:\windows\regedit.exe
0Parameters=
1CmdLine=c:\windows\system32\notepad.exe
1Parameters=
[Shutdown]
0CmdLine=D:\recent\Documents and Settings\parad0x\\1.vbs
0Parameters=
```

Содержимое файла очень просто в понимании и легко для редактирования. Файл может состоять из двух разделов — `Startup` и `Shutdown` (для файла из каталога `%systemroot%\System32\GroupPolicy\User\Scripts` это разделы `Logon` и `Logoff`). Эти разделы могут хранить записи в следующем формате:

«,»CmdLine=«»
«,»Parameters=«»

Например, в данном случае при включении компьютера будут запускаться программы `regedit.exe` и `notepad.exe` (поскольку программы запускаются до входа пользователя, вы не увидите их окон, их запуск в примере приведен для наглядности), а при завершении работы компьютера будет запускаться специальный сценарий. Вы и сами можете вручную отредактировать содержимое файлов `scripts.ini`, все внесенные вами изменения будут сразу же учтены.

Раздел Параметры безопасности элемента Конфигурация компьютера включает в себя разделы Политики учетных записей, Локальные политики, Политики открытого ключа, Политики ограниченного использования программ и Политики безопасности IP на "Локальный компьютер". Для элемента Конфигурация пользователя раздел Параметры безопасности содержит только один вложенный раздел — Политики открытого ключа, с помощью которого можно импортировать в хранилище сертификатов находящиеся на данном компьютере сертификаты. Разделы Политики учетных записей и Локальные политики являются частью оснастки Шаблоны безопасности, поэтому будут рассмотрены в разделе, описывающем данную оснастку.

Раздел Политики открытого ключа включает в себя подраздел Файловая система EFS, который позволяет создать агент восстановления данных для шифрованной файловой системы EFS. Для этого достаточно в контекстном меню подраздела Файловая система EFS выбрать команду Добавить агента восстановления данных. После этого отобразится диалоговое окно Мастера добавления агента восстановления, который перед созданием агента восстановления попросит вас указать сертификат пользователя, выступающего в роли агента восстановления.

Политика ограниченного использования программ применяется для запрещения запуска на данном компьютере тех или иных программ или для указания программ, которые запускать разрешено. При этом определение разрешенных или запрещенных для исполнения программ или сценариев возможно по четырем условиям.

- По сертификату, который выдан сценарию или пакету установщика Windows (MSI). Если сценарий имеет корректную подпись сертификата, то его запуск разрешен (или запрещен). При этом следует учитывать, что данный способ нельзя применять к файлам с расширениями EXE и DLL.

  Этот способ является наиболее защищенным способом разрешения только запуска определенных сценариев и пакетов установщика Windows, если по умолчанию запуск всех сценариев запрещен.

- По хэшу, которым подписан файл. Если для определенного файла создан хэш (последовательность байтов, гарантирующая, что данный файл не был изменен, то есть в теории не существует двух одинаковых хэшей), то на основе этого хэша можно определить, разрешено или запрещено запускать данный файл. При использовании хэша можно определять правила запуска для файлов с любым расширением.

  При определении хэша для разрешения запуска программы этот способ является хорошей альтернативой (или дополнением) способу определения разрешения запуска файла на основе подписи сертификата. Если же вы будете исполь-

зовать хэш для запрещения запуска какой-либо программы, то следует учитывать, что при изменении содержимого программы меняется и ее хэш. Другими словами, если пользователь с помощью любого компилятора изменит хотя бы один символ в файле программы, то программа будет иметь совершенно другой хэш, поэтому ее запуск будет разрешен.

- По зоне Интернета, из которой был взят пакет установщика Windows (данный способ ограничения может быть применен только к пакетам установщика Windows). По умолчанию операционная система Windows XP разделяет все пространство сети на четыре зоны: Интернет, Локальный компьютер, Местная интрасеть, Ограниченные узлы и Надежные узлы (вспомните раздел о параметрах реестра для настройки браузера Internet Explorer). На основе того, из какой зоны был взят данный пакет установщика Windows, можно определить, разрешено ли его запускать.

  Если пользователям в сети разрешено запускать не только пакеты установщика Windows, одобренные администратором (способ определения разрешения на запуск по сертификату пакета установщика Windows), то рекомендуется хотя бы настроить ограничения установки пакетов из различных зон Интернета.

- По каталогу, в котором находится файл. Можно определить каталоги, файлы, из которых запрещено или, наоборот, разрешено запускать.

Данный способ является наименее защищенным, так как его довольно легко обойти, просто переместив файл из запрещенного каталога в разрешенный. Хотя вместе с другими способами он позволяет более тонко настроить политику ограниченного использования программ.

Итак, как же все теоретические основы, описанные выше, реализуются в консоли Групповая политика? Если сказать честно, то их реализация немного запутанна. Раздел Политика ограниченного использования программ содержит два вложенных раздела (Уровни безопасности и Дополнительные правила), а также три правила: Принудительный, Назначенные типы и Доверенные издатели.

- Раздел Уровни безопасности позволяет определить основной уровень разрешения запуска программ, на котором будет работать операционная система (то есть данный уровень определяет, разрешено ли запускать файл, если его запуск не был ограничен (или разрешен) никакими политиками ограничения запуска). Основных уровней всего два — разрешать запуск всех файлов, а также запрещать запуск всех файлов. По умолчанию используется уровень разрешения запуска всех файлов, но если необходимо ограничить доступ к файлам, то рекомендуется установить основной уровень запрещения запуска всех файлов.

  Для смены основного уровня безопасности необходимо в его контекстном меню выбрать команду По умолчанию.

- Раздел Дополнительные правила. Именно этот элемент и определяет политики на запуск файлов. По умолчанию используются политики разрешения запуска файлов, расположенных в каталогах `%systemroot%`, `%systemroot%\system32`

и `%programfiles%` (способ разрешения запуска файлов по каталогу, в котором они находятся). Это необходимо для того, чтобы операционная система смогла корректно загрузиться, поэтому не рекомендуется запрещать доступ к содержимому этих каталогов.

Чтобы создать свою политику запуска файлов, необходимо воспользоваться контекстным меню раздела Дополнительные правила. Данное меню включает в себя следующие уже знакомые вам команды: Создать правило для сертификата, Создать правило для хэша, Создать правило для зоны Интернета и Создать правило для пути. При использовании создания правила для сертификата консоль управления Microsoft попросит вас указать, разрешено или запрещено запускать файлы, описанные в сертификате, а также файл сертификата, на основе которого определяется возможность доступа к файлу. При использовании создания правила для хэша консоль управления Microsoft предложит вам указать, разрешено или запрещено запускать файл, а также путь к фалу, для которого назначается политика. При использовании создания правила для зоны Интернета консоль управления Microsoft попросит вас указать, разрешено или запрещено запускать пакеты установщика Windows из зоны, а также определить зону, для которой создается данная политика. При использовании создания правила для пути консоль управления Microsoft предложит указать, разрешено или запрещено запускать файлы из каталога, а также определить сам путь к каталогу.

- Правило Принудительный позволяет определить, относятся ли к запрещенным для исполнения файлы библиотек программ (DLL), которые по умолчанию к таким файлам не относятся, а также определить, будут ли создаваемые политики ограничения доступа влиять на администраторов компьютера (по умолчанию влияют на все типы учетных записей).
- Правило Назначенные типы файлов позволяет просмотреть список всех расширений файлов, на которые распространяются политики запуска, а также добавить или удалить из данного списка определенные расширения файлов.
- Правило Доверенные издатели позволяет определить, разрешено ли обычным пользователям выбирать доверенных издателей или это разрешено только администраторам (по умолчанию это разрешено и обычным пользователям), а также позволяет указать, будет ли определяться отзыв данного сертификата по имени издателя и штампу времени (по умолчанию эти возможности отключены).

ПРИМЕЧАНИЕ

Чтобы отредактировать правило, нужно в его контекстном меню выбрать команду Свойства.

Теперь рассмотрим ветви реестра, на которые влияют политики ограничения запуска программ. Все настройки политик ограничения запуска программ находятся в ветви реестра `HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Microsoft\Windows\Safer\CodeIdentifiers`. Ветвь содержит следующие параметры.

ПРИМЕЧАНИЕ

Для хранения параметров политик ограничения доступа также используются разделы из ветви реестра HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Group Policy Objects\{GUID-номер объекта групповой политики}Machine\Software\Policies\Microsoft\Windows\Safer\Codeidentifiers. Разделы из этой ветви имеют больший приоритет, поэтому желательно редактировать именно их.

- `DefaultLevel` — этот параметр `DWORD`-типа определяет основной режим безопасности компьютера. По умолчанию значение равно `0õ00040000`, оно показывает, что разрешен запуск любых файлов. Если же значение данного параметра равно `0`, то по умолчанию запуск файлов будет запрещен.
- `ExecutableTypes` — данный параметр `REG_MULTI_SZ`-типа содержит список всех расширений файлов, на которые будут действовать политики ограничений запуска программ.
- `LogFileName` — параметр строкового типа, хранит путь к текстовому файлу (и его название), в который будут заноситься сведения обо всех попытках запуска запрещенных и разрешенных программ. По умолчанию параметр отсутствует.
- `PolicyScope` — если значение данного параметра `DWORD`-типа равно `1`, то политики ограничений запуска не будут относиться к администраторам локального компьютера. Если же значение данного параметра равно `0`, то политики ограничений будут накладываться на все учетные записи данного компьютера.
- `TransparentEnabled` — если значение данного параметра `DWORD`-типа равно `2`, то политики ограничений запуска будут относиться не только к программам, но и к библиотекам DLL, которые используются этими программами. Если же значение равно `1`, то ограничения не будут накладываться на библиотеки, используемые запрещенными программами.

Сами же настройки политик ограничения располагаются в одном из разделов ветви системного реестра `HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Group Policy Objects\{GUID-íîìåð îáúåêòà ãðóïïîâîé ïîëèòèêè}Machine\Software\Policies\Microsoft\Windows\Safer\CodeIdentifiers` (по умолчанию отсутствует). Данная ветвь реестра включает в себя два раздела: `0` и `262144`. Раздел `0` хранит стандартные политики по ограничению доступа (точнее, по разрешению) к файлам в каталогах `%systemroot%`, `%systemroot%\system32` и `%programfiles%`. Раздел `262144` содержит пользовательские политики ограничения запуска файлов. Оба этих раздела могут хранить следующие вложенные подразделы.

- `Hashes` — содержит политики ограничения доступа к файлам по их хэшу. Для каждой политики по ограничению файла в этом подразделе находится еще два подраздела. Первый из подразделов определяет хэш файла MD-5, а второй — хэш файла по алгоритму SHA-1.

- `Paths` — хранит политики ограничения доступа к файлам по каталогу, в котором они хранятся.
- `UrlZones` — содержит политики ограничения доступа к пакетам установщика Windows взятым из определенной зоны Интернета.

Политика безопасности IP позволяет настроить протокол IPSec для защиты пакетов, передаваемых между компьютерами, на тот или иной уровень безопасности. При этом стоит учитывать, что для работы протокола IPSec необходима служба Службы IPSEC и если эта служба остановлена, то вы не сможете воспользоваться возможностью защиты передаваемых данных с помощью протокола IPSec. По умолчанию доступны только три политики безопасности IP — Клиент (Ответ только), Сервер (Запрос безопасности) и Сервер безопасности (Требуется безопасность). Все эти политики для обеспечения подлинности используют протокол Kerberos, то есть их нельзя применять для компьютеров, не входящих в домен Active Directory (хотя с помощью диалога Свойства метод проверки подлинности можно изменить). Вместо изменения стандартных политик рекомендуется создать свои собственные. Для этого достаточно в контекстном меню раздела Политика безопасности IP на "Локальный компьютер" выбрать команду Создать политику безопасности IP. После этого отобразится окно Мастера политики IP-безопасности, который предложит вам задать имя политики, ее описание, а также указать, будет ли данная политика использоваться по умолчанию для установки соединений с другими компьютерами. Если данная политика не будет использоваться по умолчанию, то мастер закончит свою работу и выведет диалог Свойства созданной вами политики, чтобы вы могли ее настроить. Если же политика будет использоваться по умолчанию, то мастер попросит вас также указать метод проверки подлинности, используемый по умолчанию для установки соединения. Возможны три метода: метод с использованием протокола Kerberos, метод с использованием сертификатов и метод с использованием пароля (общий секрет). После того как вы укажете метод проверки подлинности при установке соединения, мастер закончит свою работу и выведет диалог Свойства созданной вами политики.

В контексте данной книги диалог Свойства политик безопасности IP рассмотрен не будет, так как для полного и понятного описания настроек этих политик может понадобиться отдельная книга.

С помощью раздела Настройка Internet Explorer можно настроить интерфейс браузера Internet Explorer, а также параметров его подключения к Интернету. Этот раздел консоли Групповая политика содержит следующие вложенные разделы: Пользовательский интерфейс обозревателя, Подключение, URL-адреса, Безопасность и Программы. Вкратце рассмотрим возможности, которые предоставляют эти разделы.

Раздел Пользовательский интерфейс обозревателя позволяет настроить такие элементы окна браузера Internet Explorer, как заголовок окна (Заголовок обозревателя), изображения для фона панели инструментов (Настройка панели инструментов), а также логотипы браузера (можно настраивать как статический логотип, так и GIF-файл для отображения динамического логотипа при подключении к сайту) (Эмблемы).

Можно также добавить собственные кнопки к панели инструментов. Раньше уже были описаны возможности изменения всех этих элементов интерфейса с помощью реестра — эти же параметры реестра применяются и консолью управления Microsoft, хотя при их изменении с ее помощью есть и некоторые очень интересные особенности, которые будут описаны ниже.

Раздел Подключение позволяет настроить такие параметры браузера, как строка, добавляемая к строке обозревателя (Строка обозревателя), используемые для подключения к прокси-серверу адреса и порты (для каждого протокола) (Параметры прокси-сервера), а также адрес компьютера, содержащего сценарий для автоматической настройки обозревателя (Автоматическая настройка обозревателя). С помощью данного раздела можно также импортировать настройки, расположенные на вкладке Подключения диалога Свойства обозревателя, в файлы, расположенные в каталоге `%systemroot%\system32\GroupPolicy\User\MICROSOFT\IEAK\BRANDING\cs`.

Раздел URL-адреса позволяет настроить содержимое папок Избранное и Ссылки (Избранное и ссылки), а также задать стандартные адреса Интернета (Важные URL-адреса). Под стандартными адресами понимаются следующие: адрес домашней страницы, адрес панели поиска и адрес страницы поддержки.

Раздел Безопасность позволяет импортировать настройки зон Интернета и настройки ограничений браузера в INF-файлы (Зоны безопасности и оценка содержимого). Зоны безопасности импортируются в файлы `seczones.inf` и `seczrsop.inf` каталога `%systemroot%\system32\GroupPolicy\User\MICROSOFT\IEAK\BRANDING\ZONES`, а параметры оценки содержимого (Rating) импортируются в INF-файлы `ratings.inf` и `ratrsop.inf`, которые расположены в каталоге `%systemroot%\system32\GroupPolicy\User\MICROSOFT\IEAK\BRANDING\RATINGS`. С помощью этого раздела можно импортировать параметры настройки Authenticode (сертификаты доверенных издателей, а также сертификаты доверенных агентств выдачи сертификатов). Для этого предназначен элемент Параметры Authenticode.

Раздел Программы позволяет импортировать настройки вкладки Программы, расположенной в диалоговом окне Свойства обозревателя, в файл `programs.inf`. Этот файл находится в каталоге `%systemroot%\system32\GroupPolicy\User\MICROSOFT\IEAK\BRANDING\PROGRAMS`.

Теперь поговорим о том, как выполняется запись в реестр всех настроек, расположенных в разделе Настройка Internet Explorer. По умолчанию все настройки из этого раздела заносятся в файл `install.ins`, расположенный в каталоге `%systemroot%\system32\GroupPolicy\User\MICROSOFT\IEAK`. По умолчанию только администраторы могут выполнять запись данных в этот каталог, хотя модифицировать файл `install.ins` можно от имени любого пользователя. Это обычный текстовый файл с расширением INS, хранящий настройки, которые можно изменить с помощью раздела Настройка Internet Explorer. Данный файл довольно прост в понимании, поэтому не будем останавливаться на описании каждого его раздела,

а просто приведем листинг содержимого этого файла (где это было возможно, адреса и названия создаваемых элементов описывают сами создаваемые элементы).

Листинг 11.10. Содержимое файла install.ins

```
[Branding]
GPVersion=6.00.2900.2180
NoFavorites=1
NoLinks=1
Window_Title_CN=Заголовок окна
Window_Title=Microsoft Internet Explorer предоставлено: Ваша компания
ToolbarBitmap=E:\images\fotco\Art_galery\Название_фирмы_слоган_фирмы_Название_фирмы.bmp
UserAgent=Название_фирмы/Название_фирмы_слоган_фирмы_Название_фирмы
[Small_Logo]
Name=Название_фирмы_22_22.BMP
Path=D:\my_books\Название_фирмы_22_22.BMP
[Big_Logo]
Name=Название_фирмы_38_38.BMP
Path=D:\my_books\Название_фирмы_38_38.BMP
[Animation]
Small_Name=Название_фирмы_слоган_фирмы_Название_фирмы_22_22.BMP
Small_Path=D:\my_books\Название_фирмы_слоган_фирмы_Название_фирмы_22_22.BMP
Big_Name=Название_фирмы_слоган_фирмы_Название_фирмы_38_38.BMP
Big_Path=D:\my_books\Название_фирмы_слоган_фирмы_Название_фирмы_38_38.BMP
DoAnimation=1
[URL]
Search_Page=http://www.Название_фирмы_Название_фирмы_Название_фирмы.com
AutoConfigTime=5
AutoDetect=1
AutoConfig=1
AutoConfigURL=http://www.Название_фирмы_Название_фирмы_слоган_фирмы_INS_Название_фирмы.ru
AutoConfigJSURL=http://www.Название_фирмы_Название_фирмы_слоган_фирмы_JS_Название_фирмы_и_Название_фирмы.ru
Home_Page=http://www.Название_фирмы_Название_фирмы_Название_фирмы.com
Help_Page=http://www.Название_фирмы_Название_фирмы_Название_фирмы.com
[ExtRegInf]
SecZones=*,seczones.inf,DefaultInstall
Ratings=*,ratings.inf,DefaultInstall
Programs=*,programs.inf,DefaultInstall
connset=connect.inf,DefaultInstall
```

```
[ExtRegInf.Hkcu]
connset=connect.inf,IeakInstall.Hkcu
SecZones=seczones.inf,IeakInstall.Hkcu
Programs=*,programs.inf,IEAKInstall.HKCU
[Proxy]
HTTP_Proxy_Server=666.66.66.66:80
FTP_Proxy_Server=666.66.66.66:80
Gopher_Proxy_Server=666.66.66.66:80
Secure_Proxy_Server=666.66.66.66:80
Socks_Proxy_Server=666.66.66.66:80
Proxy_Override=<local>
Use_Same_Proxy=0
Proxy_Enable=1
[Security Imports]
TrustedPublisherLock=0
ImportSecZones=1
ImportRatings=1
[ExtRegInf.Hklm]
SecZones=seczones.inf,IeakInstall.Hklm
Ratings=ratings.inf,IEAKInstall.HKLM
Programs=*,programs.inf,IEAKInstall.HKLM
[BrowserToolbars]
Caption0=□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□
Action0=D:\Program files\Filemon.exe
Icon0=E:\images\samplies.ico
HotIcon0=E:\images\samplies.ico
Show0=1
[ConnectionSettings]
ConnectName0=Nokia 7270 USB (OTA)
ConnectSize0=2884
ConnectName1=Nokia 7270 USB (OTA)
ConnectSize1=2884
Option=1
EnableAutodial=1
NoNetAutodial=1
```

Файл хранит большое количество настроек, а также ссылок на другие INF-файлы, импортированные с помощью раздела Настройка Internet Explorer. Все импортируемые файлы, на которые содержатся ссылки в описываемом файле, экспортируются

в реестр при открытии любого элемента раздела Настройка Internet Explorer. При этом же в реестр заносятся все настройки из файла `install.ins`. Причем самое главное состоит в том, что эти настройки заносятся не консолью управления Microsoft, как можно было бы подумать, а процессом `WINLOGON.EXE`, запущенным от имени системы. Другими словами, даже если пользователь не обладает правами на изменение указанных в файле `install.ins` ветвей реестра, они все равно будут изменены.

ПРИМЕЧАНИЕ

Довольно странное решение Microsoft. В конце книги я попытаюсь высказать свои суждения по поводу использования запущенных от имени системы процессов в общем и о данном способе записи содержимого файла в реестр в частности.

Административные шаблоны

С помощью административных шаблонов можно более тонко настроить как ограничения на работу отдельных компонентов компьютера, так и сами эти компоненты. Многие считают, что административные шаблоны и являются групповой политикой, хотя это не совсем так. Если учесть, что элементы групповой политики используют для своей реализации ветви системного реестра Windows `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\policies` и `HKEY_LOCAL_MACHINE\SOFTWARE\Policies` (а также эти ветви из корневого раздела `HKEY_CURRENT_USER`), то к групповым политикам можно отнести большинство элементов одноименной консоли. Особенностью административных шаблонов является то, что все их настройки берутся из специальных текстовых файлов с расширением ADM. Эти файлы написаны на специальном языке сценариев и расположены в каталоге `%systemroot%\system32\GroupPolicy\Adm`. Копии этих файлов могут находиться в каталоге `%systemroot%\inf` (причем, как правило, в этом каталоге находятся дополнительные ADM-файлы, не принимающие участия в построении списка Административные шаблоны). Вкратце рассмотрим назначение всех этих ADM-файлов.

- `system.adm` — имеет размер около 1824 Кбайт и хранит большую часть настроек конфигурации различных компонентов компьютера. По умолчанию он используется для построения элемента Административные шаблоны.
- `wuau.adm` — размером около 44 Кбайт и содержит настройки ограничений на работу автоматического обновления Windows. По умолчанию используется для построения элемента Административные шаблоны.
- `wuau.adm` — имеет размер около 44 Кбайт и хранит настройки ограничений на работу автоматического обновления Windows. По умолчанию он используется для построения элемента Административные шаблоны.
- `wmplayer.adm` — размером около 69 Кбайт и содержит настройки ограничений на работу Проигрывателя Windows Media. По умолчанию используется для построения элемента Административные шаблоны.

- `conf.adm` — имеет размер около 42 Кбайт и хранит настройки ограничений на работу программы NetMeeting. По умолчанию он используется для построения элемента Административные шаблоны.
- `inetres.adm` — размером около 1470 Кбайт и содержит настройки ограничений на работу браузера Internet Explorer. По умолчанию используется для построения элемента Административные шаблоны.
- `inetset.adm` — имеет размер около 17 Кбайт и хранит дополнительные настройки ограничений на работу браузера Internet Explorer. Он не используется для построения элемента Административные шаблоны. Кроме того, следует учитывать, что использование возможностей данного файла оставляет «татуировки на реестре». Об этом термине будет рассказано чуть позже.
- `inetcorp.adm` — размером около 7 Кбайт и содержит дополнительные настройки ограничений на работу браузера Internet Explorer. Он не используется для построения элемента Административные шаблоны. Кроме того, следует учитывать, что использование возможностей данного файла оставляет «татуировки на реестре». А главное, надо учесть, что данный файл был написан для более ранних версий операционной системы Windows, чем Windows 2000.

Обычно, каждое правило административных шаблонов может иметь три состояния.

- Не задан — если правило имеет данное значение, то параметр реестра, используемый этим правилом, удален из реестра. Иными словами, если присвоить правилу данное состояние, то параметр, используемый правилом, будет удален из реестра. Данное состояние корректно обрабатывается стандартными ADM-файлами Windows XP. Если же оно не будет поддерживаться правилами ADM-файла (как это было в Windows NT), то данные правила будут оставлять «татуировки на реестре». Другими словами, установка состояния Не задан не будет приводить к удалению параметра, и этот параметр и дальше будет ограничивать правило.
- Включен — если правило имеет данное состояние, то значение параметра реестра, используемого этим правилом, будет равно `1`.
- Отключен — если правило имеет данное состояние, то значение параметра реестра, используемого этим правилом, будет равно `0`.

Здесь не будут рассмотрены все существующие правила элемента Административные шаблоны, так как они и так содержат подробное объяснение того, что они делают. Чтобы просмотреть это объяснение, достаточно выбрать правило и перейти на расширенный вид (рис. 11.9) либо в диалоге Свойства конкретного правила (открывается выбором команды Свойства из контекстного меню правила) выбрать вкладку Объяснение. На вкладке Параметры можно узнать версии Windows, для которых применяется данное правило.

Тем не менее, чтобы иметь общее представление о том, что позволяют сделать административные шаблоны, рассмотрим некоторые из их правил.

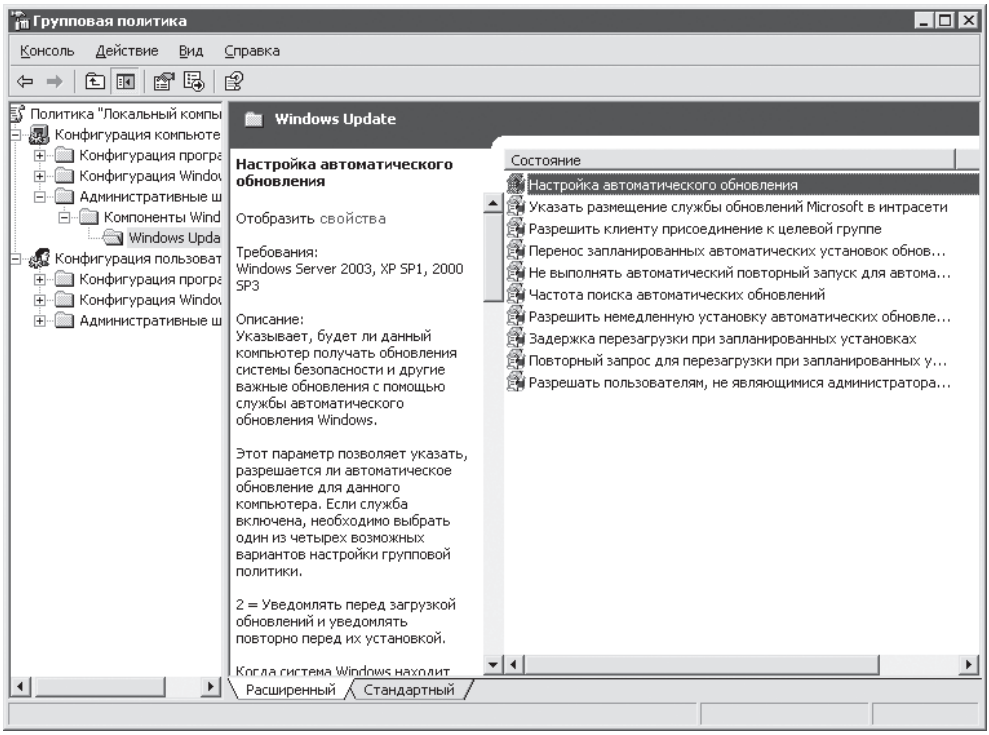


Рис. 11.9. Описание правила

Компоненты Windows ▸ Windows Update. Раздел находится в элементе Конфигурация компьютера и строится на основе ADM-файла `wuau.adm`. Раздел хранит настройки автоматического обновления и использует для своей работы ветвь реестра `HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Microsoft\Windows\WindowsUpdate\AU`. Эта ветвь может содержать следующие параметры DWORD-типа.

- `NoAutoUpdate` — если значение равно `1`, то возможность автоматического обновления будет отключена.
- `AUOptions` — определяет режим работы автоматического обновления (если значение параметра `NoAutoUpdate` равно `0`). Он может принимать следующие значения:
  - `2` — перед загрузкой обновлений и перед установкой уведомлений система будет выдавать сообщение об этом;
  - `3` — система будет автоматически загружать обновления, но перед их установкой будет уведомлять пользователя (данный режим используется по умолчанию);
  - `4` — система будет как загружать, так и устанавливать обновления без уведомления пользователя, причем устанавливать будет лишь в то время, которое указано в параметрах `ScheduledInstallDay` и `ScheduledInstallTime`;

  - 5 — если параметр будет иметь это значение, то системному администратору предстоит самому решить, какой режим работы автоматического обновления должен использоваться.

- ScheduledInstallDay — определяет день, в который будут устанавливаться все скачанные обновления, если значение параметра AUOptions равно 4. Скачанные обновления будут устанавливаться:
  - 0 — ежедневно (данное значение используется по умолчанию);
  - 1 — каждое воскресенье;
  - 2 — каждый понедельник;
  - 3 — каждый вторник;
  - 4 — каждую среду;
  - 5 — каждый четверг;
  - 6 — каждую пятницу;
  - 7 — каждую субботу.
- ScheduledInstallTime — определяет время, в которое будут устанавливаться все скачанные обновления, если значение параметра AUOptions равно 4.

С помощью административных шаблонов можно указать компьютер, с которого будут скачиваться обновления, и многие другие параметры работы автоматического обновления. Более подробно об этом можно прочитать в описаниях правил административных шаблонов или в базе данных реестра, поставляемой вместе с этой книгой.

Компоненты Windows ▸ Проигрыватель Windows Media. Раздел находится в элементе Конфигурация компьютера и строится на основе ADM-файла wmplayer.adm. Раздел содержит настройки конфигурации Проигрывателя Windows Media и использует для работы ветвь реестра HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Microsoft\WindowsMediaPlayer, которая может хранить следующие параметры.

- DontUseFrameInterpolation — определяет, будет ли использоваться сглаживание изображения при просмотре видео. Сглаживание изображения на мощных компьютерах может вызвать повышение качества, но на маломощных может привести лишь к притормаживанию и артефактам изображения. Если значение данного параметра равно 1, то возможность сглаживания изображений использоваться не будет.
- DisableAutoUpdate — если значение равно 1, то возможность автоматического обновления проигрывателя будет отключена.

Кроме того, с помощью данного пути в элементе Конфигурация пользователя можно скрыть следующие вкладки диалога Параметры: Сеть, Конфиденциальность и Безопасность, а также настроить многие другие параметры интерфейса проигрывателя. Более подробно об этом можно прочитать в описаниях правил административных шаблонов или в базе данных реестра, поставляемой вместе с книгой.

Компоненты Windows ▸ Internet Explorer ▸ Панель управления обозревателем. Раздел находится как в элементе Конфигурация компьютера, так и в элементе Конфигурация пользователя, и строится на основе ADM-файла `inetres.adm`. Раздел содержит настройки конфигурации диалога Свойства обозревателя и использует для работы ветвь реестра `HKEY_CURRENT_USER\Software\Policies\Microsoft\Internet Explorer\Control Panel` (или ветвь из корневого раздела `HKEY_LOCAL_MACHINE`), которая может хранить следующие параметры `DWORD`-типа.

- `Privacy Settings` — если значение равно `1`, то будет запрещено изменять настройки параметров на вкладке Конфиденциальность.
- `PrivacyTab` — при установке значения равным `1` в диалоге Свойства обозревателя будет скрыта вкладка Конфиденциальность.

ВНИМАНИЕ

Несмотря на все приведенные выше запреты, диалоговое окно Параметры блокирования всплывающих окон можно будет запустить с помощью команды rundll32.exe inetcpl.cpl, DisplayPopupWindowManagementDialog.

- `ConnectionsTab` — если значение равно `1`, то в диалоге Свойства обозревателя будет скрыта вкладка Подключение.
- `Connection Settings` — при установке значения равным `1` будет запрещено изменение содержимого на вкладке Подключение.
- `Autoconfig` — если значение равно `1`, то флажок Использовать сценарий автоматической настройки в диалоговом окне Настройка локальной сети будет снят, а поле Адрес данного диалогового окна будет неактивно (окно отображается после нажатия кнопки Настройка LAN, расположенной на вкладке Подключения диалога Свойства обозревателя).

ВНИМАНИЕ

Несмотря на все приведенные выше запреты, флажок Использовать сценарий автоматической настройки (как и поле Адрес) можно будет установить с помощью команды rundll32.exe INETCFG.dll, InetSetAutoProxyA «URL или IP-адрес компьютера, содержащего сценарий настройки».

- `ProgramsTab` — при установке значения равным `1` в диалоге Свойства обозревателя будет скрыта вкладка Программы.
- `AdvancedTab` — если значение равно `1`, то в диалоге Свойства обозревателя будет скрыта вкладка Дополнительно.
- `Advanced` — при установке значения равным `1` будет запрещено редактирование параметров на вкладке Дополнительно диалога Свойства обозревателя.
- `GeneralTab` — если значение равно `1`, то в диалоге Свойства обозревателя будет скрыта вкладка Общие.

ВНИМАНИЕ

Несмотря на приведенный выше запрет, диалог Языки, который можно было открыть с помощью вкладки Общие, можно будет запустить с помощью команды rundll32.exe inetcpl.cpl, OpenLanguageDialog.

- `SecurityTab` — при установке значения равным 1 в диалоге Свойства обозревателя будет скрыта вкладка Безопасность.
- `SecAddSites` — если значение равно 1, то будет запрещено изменение узлов в зонах, отображаемых при нажатии кнопки Узлы на вкладке Безопасность.
- `SecChangeSettings` — при установке значения равным 1 будет запрещено изменять уровни безопасности на вкладке Безопасность диалога Свойства обозревателя.

ВНИМАНИЕ

Несмотря на все приведенные выше запреты, диалог Безопасность можно будет запустить с помощью команды rundll32.exe inetcpl.cpl, LaunchSecurityDialogEx.

- `ContentTab` — если значение равно 1, то в диалоге Свойства обозревателя будет скрыта вкладка Содержание.
- `Ratings` — при установке значения равным 1 будет запрещено изменять ограничение доступа к страницам Интернета с помощью вкладки Содержание.
- `Certificates` — если значение равно 1, то кнопки Сертификаты и Издатели на вкладке Содержание будут неактивны.
- `CertifPers` — при установке значения равным 1 кнопка Сертификаты на вкладке Содержание будет неактивна.
- `CertifPub` — если значение равно 1, то кнопка Издатели на вкладке Содержание будет неактивна.
- `CertifSite` — при установке значения равным 1 кнопка Сертификаты на вкладке Содержание будет неактивна.

ВНИМАНИЕ

Несмотря на все приведенные выше запреты, диалог Сертификаты (как и диалог Издатели) можно будет запустить с помощью команды rundll32.exe CRYPTUI.dll, CryptUIStartCertMgr или других подобных ей команд, которые были рассмотрены в части 1.

Несмотря на все приведенные выше запреты, диалог Ограничение доступа можно будет запустить с помощью команды rundll32.exe IEAKENG.dll, ModifyRatings или других подобных ей команд, которые были рассмотрены в части 1. Можно также включить ограничение доступа. Для этого необходимо воспользоваться командой rundll32.exe MSRATING.dll, RatingEnable.

Как можно заметить, многие из приведенных ограничений довольно просто обходятся с помощью команд `rundll32` (именно поэтому они были описаны), поэтому эти ограничения использовать не рекомендуется. Вообще, если безопасность компьютера имеет первостепенную важность, то рекомендуется не полагаться на все ограничения, которые просто скрывают вкладки или делают неактивными те или иные элементы, так как такие ограничения в большинстве случаев можно обойти или с помощью команд `rundll32`, или непосредственно с помощью редактирования реестра.

ПРИМЕЧАНИЕ

Если же вы все-таки решили использовать возможности скрытия вкладок диалога Свойства обозревателя, то могу предложить интересный способ их быстрой установки. Для этого нужно вызвать команду rundll32.exe IEAKENG.dll, ShowInetcpl. После выполнения этой команды rundll32 сам создаст все приведенные выше параметры и присвоит им значение 1. Затем он попытается открыть диалог Свойства обозревателя, после чего напишет, что открытие данного диалога запрещено администратором. Если нажать одну из кнопок данного диалога, то все созданные параметры ограничений будут удалены, но если просто завершить процесс rundll32 (с помощью Диспетчера задач), то параметры так и останутся установленными в реестре.

Можно также воспользоваться командой rundll32.exe IEAKENG.dll, ModifyZones для удаления параметров скрытия вкладок Безопасность и Конфиденциальность из корневого раздела HKEY_CLASSES_ROOT. После выполнения данной команды параметры PrivacyTab и SecurityTab будут удалены из реестра, а при закрытии диалога Свойства обозревателя снова созданы.

Панель управления. Раздел находится в элементе Конфигурация пользователя и строится на основе ADM-файла `system.adm`. С его помощью можно запретить запуск Панели управления, а также всех CPL-файлов, установленных на компьютере. При попытке открытия этих файлов система будет писать о том, что это запрещено администратором. Тем не менее в диалоге с помощью команд `rundll32` можно обойти данное ограничение на некоторые CPL-файлы. Например, следующие команды открывают апплеты:

- `rundll32.exe Access.cpl, DebugMain` — апплет Специальные возможности (`Access.cpl`);
- `rundll32.exe firewall.cpl, ShowControlPanel` — Брандмауэр Windows (`firewall.cpl`);
- `rundll32.exe joy.cpl, ShowJoyCPL` — Игровые устройства (`joy.cpl`);
- `rundll32.exe mmsys.cpl, ShowFullControlPanel` — апплет Свойства: Звук и аудиоустройства (`mmsys.cpl`), который также можно вызвать с помощью команд `rundll32.exe mmsys.cpl, ShowDriverSettingsAfterFork` и `rundll32.exe mmsys.cpl, ShowAudioPropertySheet`;
- `rundll32.exe netplwiz.dll, UsersRunDll` — Учетные записи пользователей;

- `rundll32.exe newdev.dll, WindowsUpdateDriverSearchingPolicyUi` — несмотря на запрет на доступ к апплету Свойства системы, открывает диалог Подключение к Windows Update;
- `rundll32.exe TAPI32.dll, internalConfig` — апплет Телефон и модем (`telephon.cpl`), который можно также вызвать с помощью следующей команды: `rundll32.exe TAPI32.dll, LOpenDialAsst`;
- `rundll32.exe wuaucpl.cpl, ShowAUControlPanel` — апплет Автоматическое обновление (`wuaucpl.cpl`).

Как можно заметить, список таких CPL-файлов, которые можно запустить, несмотря на запрет запуска, довольно велик. По этой причине рекомендуется не использовать данное ограничение административных шаблонов, а напрямую запрещать полный доступ (оставив только доступ на чтение) к ветвям реестра, параметры из которых используются CPL-файлами (как правило, это разделы ветви `HKEY_CURRENT_USER\Control Panel`).

Теперь рассмотрим некоторые другие ограничения административных шаблонов, которые можно обойти с помощью команд `rundll32`.

- Панель управления ▸ Установка и удаление программ — здесь содержатся правила, с помощью которых можно скрыть вкладку Замена или удаление программы или сам апплет Установка и удаление программ. Тем не менее удалить программу все еще можно будет с помощью рассмотренной команды `rundll32.exe appwiz.cpl, WOW64Uninstall_RunDLL ,,,«ïîâêàòàëîã ïðîãðàìì»`.

  Несмотря на то, что есть запрет скрытия вкладки Добавление и удаление компонентов Windows, с помощью следующей команды: `rundll32.exe netshell.dll, HrLaunchNetworkOptionalComponents` — можно будет отобразить диалоговое окно Мастер дополнительных сетевых компонентов Windows.
- Система ▸ Управление связью через Интернет ▸ Параметры связи через Интернет — здесь находятся правила, с помощью которых можно запретить отображение Мастера веб-публикаций. Тем не менее данный мастер все равно можно будет вызвать с помощью команды `rundll32.exe NETPLWIZ.dll, PublishRunDll` (но только для публикации содержимого папки Мои документы).
- Сеть ▸ Сетевые подключения — здесь можно запретить доступ к Мастеру новых подключений. Тем не менее данный мастер все равно можно будет вызвать с помощью команды `rundll32.exe netshell.dll, StartNCW`.
- Компоненты Windows ▸ Проводник — здесь можно скрыть диалоги подключения и отключения сетевых дисков. Тем не менее вызвать данные диалоги все-таки будет можно. Для этого соответственно применяются команды `rundll32.exe shell32.dll, SHHelpShortcuts_RunDLL Connect` и `rundll32.exe shell32.dll, SHHelpShortcuts_RunDLL Disconnect`.

Рассмотрим два ADM-файла.

- `Inetcorp.adm` — раньше было сказано, что настройки из данного файла использовались лишь в операционных системах более ранних версий, чем операционная

система Windows 2000. Возможно, это не совсем так, поэтому рассмотрим некоторые из параметров реестра, которые изменяются с помощью данного файла. Например, в реестре до сих пор присутствует DWORD-параметр `CacheLimit`, расположенный в ветви реестра `HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Internet Settings\Cache\Content`. Как говорится в оснастке, данный параметр определяет максимальный размер (в килобайтах), резервируемый для временных файлов Интернета (пользовательского кэша). А в ветви реестра `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings` содержится параметр строкового типа `CodeBaseSearchPath`, который, как говорится в файле, хранит путь для поиска документов.

- `Inetset.adm` — несмотря на то, что данный ADM-файл по умолчанию не загружается в административные шаблоны, параметры ограничений, записанные в нем, поддерживаются операционной системой Windows XP. Единственным минусом файла является то, что он оставляет «татуировки на реестре». Тем не менее в образовательных целях рассмотрим структуру этого ADM-файла. Как правило, по умолчанию данный ADM-файл находится в каталоге `%systemroot%\inf`, но если там этого файла нет, то его можно взять из установочного диска Windows XP. Он расположен в каталоге `I386` диска и называется `INETSET.AD_`. По умолчанию все файлы операционной системы на установочном диске для экономии места хранятся в сжатом виде, поэтому для распаковки данного файла необходимо воспользоваться командой `expand «ïóòü ê ôàéëó è åãî èìÿ» «ïóòü ê êàòàëîãó, â êîòîðûé ôàéë áóäåò ðàñïàêîâàí»`. При этом перед распаковкой рекомендуется скопировать распаковываемый файл в другой каталог. Можно также распаковать несколько файлов, например команда `expand i:\i386\*.ad_ d:\` распакует на диск `D:` все файлы с расширением AD_, расположенные в каталоге `i:\i386`. После распаковки файла его расширение AD_ нужно заменить расширением ADM.

Чтобы подключить любой сторонний административный шаблон, необходимо в контекстном меню элемента Административные шаблоны выбрать команду Добавление и удаление шаблонов. После ее вызова перед вами отобразится список уже загруженных ADM-файлов, из которого можно удалить одни файлы (кнопка Удалить) или добавить другие (кнопка Добавить). Для добавления ADM-файла перед запуском консоли Групповая политика также достаточно скопировать ADM-файл в каталог `%systemroot%\system32\grouppolicy\Adm` — все ADM-файлы из этого каталога загружаются в элемент Административные шаблоны по умолчанию.

После загрузки файла `inetset.adm` в элемент Административные шаблоны в нем появятся следующие разделы: Автозаполнение, Отображать параметры, Дополнительные параметры, Кодирование URL. Они расположены в элементе Конфигурация пользователя, а в элементе Конфигурация компьютера расположен только один новый раздел — Обновления компонентов. По умолчанию все они пусты. Это и неудивительно, ведь данные административные шаблоны создавались не для Windows XP и делают «татуировки на реестре», поэтому по умолчанию содержимое данных разделов отфильтровывается. Чтобы отобразить диалоговое окно настройки фильтрации, нужно в контекстном меню элемента Административные шаблоны выбрать команду

Вид ▸ Фильтрация. После этого перед вами отобразится диалог, с помощью которого можно запретить отображение правил, не удовлетворяющих какой-либо версии Windows. В данном случае в диалоге нужно снять флажок Показывать только управляемые параметры политики (это нужно делать отдельно для элемента Конфигурация компьютера и отдельно для элемента Конфигурация пользователя). После этого все правила, доступные с помощью файла `inetres.adm`, отобразятся в описанных выше разделах. Здесь не будут рассмотрены названия этих правил — рассмотрим только параметры реестра, которые изменяются этими правилами.

- `HKEY_CURRENT_USER\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\AutoComplete` — хранит настройки автозаполнения и может содержать следующие параметры строкового типа.
  - `Append Completion` — если значение равно `no`, то возможность автозаполнения для веб-адресов будет отключена. Если же значение равно `yes`, то включена.
  - `Use AutoComplete` — при установке значения равным `no` возможность автозаполнения в Проводнике Windows будет запрещена. Если же значение равно `yes`, то разрешена.
  - `AutoSuggest` — если значение равно `no`, то возможность автозаполнения для адресов Сети будет запрещена. Если же значение равно `yes`, то разрешена.
- `HKEY_CURRENT_USER\Software\Microsoft\Internet Explorer\Main` — содержит основные настройки браузера Internet Explorer. Файл `inetset.adm` позволяет редактировать значения следующих параметров: `Use FormSuggest`, `FormSuggest Passwords`, `FormSuggest PW Ask`, отвечающих за возможности использования автозаполнения форм. За более детальной информацией обращайтесь к базе данных по реестру, поставляемой вместе с книгой.

  Но, кроме этих параметров, данный файл определяет и другие.
  - `Use_DlgBox_Colors` — если значение равно `yes`, то при отображении сайтов будут использоваться цвета Windows.
  - `Disable Script Debugger` — при установке значения равным `yes` браузер не будет разрешать отладку сценария при ошибке в нем.
  - `Error Dlg Displayed On Every Error` — если значение равно `yes`, то браузер будет отображать сообщение об ошибке при каждой ошибке сценария.
- `HKEY_CURRENT_USER\Software\Microsoft\Internet Explorer\Security\P3Global` — если данная ветвь реестра будет содержать `DWORD`-параметр `Enabled`, равный `1`, то редактор профиля будет задействован.
- `HKEY_CURRENT_USER\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings` — хранит следующие рассматриваемые в файле параметры `DWORD`-типа.
  - `WarnOnPostRedirect` — если значение равно `1`, то при перенаправлении передаваемой пользователем формы браузер будет предупреждать пользователя.

  - `WarnOnZoneCrossing` — при установке значения равным 1 браузер будет предупреждать пользователя о переключении зоны безопасности.
  - `UrlEncoding` — если значение равно 1, то будет задействована возможность посылки адресов в формате UTF8.
- `HKEY_CURRENT_USER\Software\Microsoft\Java VM` — хранит настройки консоли Java (помните рассмотренный ранее параметр `EnableJavaConsole`?). С помощью данного файла можно отредактировать следующие параметры `BINARY`-типа.
  - `EnableLogging` — если значение равно 1, то возможность протоколирования виртуальной машины Java от Microsoft будет включена.
  - `EnableJIT` — при установке значения равным 1 будет включен компилятор Microsoft Virtual Machine JIT.
- `HKEY_CURRENT_USER\Software\Microsoft\Internet Explorer\SearchUrl` — если значение параметра строкового типа `provider` из данной ветви реестра равно `INTRANET`, то будет использоваться внутренний сервер автопоиска (о настройке возможности автопоиска уже было сказано в разделе о параметрах реестра, предназначенных для настройки браузера Internet Explorer).

ПРИМЕЧАНИЕ

Описываемый ADM-файл содержит и другие настройки. Эти настройки не были рассмотрены по той причине, что их можно установить и с помощью стандартных диалогов браузера.

На этом будет закончено рассмотрение групповых политик в целом и административных шаблонов в частности. Тем не менее хотелось бы напомнить, что те параметры административных шаблонов, которые были рассмотрены, являются лишь каплей в море возможностей настройки компонентов Windows XP, предоставляемых административными шаблонами. Просто представьте, сколько сотен параметров описано только в двух единственных ADM-файлах `system.adm` и `inetres.adm`, если учесть, что они имеют размеры по 1500 Кбайт каждый.

Результирующая политика

Результирующая политика — это новый механизм Windows XP, позволяющий просмотреть общие настройки групповой политики для конкретного пользователя или конкретного компьютера. На данный момент групповые политики содержат около 1000 различных параметров реестра, которые могут перекрывать действия друг друга. Но это не главное. Наиболее сильно проблема усугубляется в домене Active Directory. Стоит лишь вспомнить, что групповые политики для

компьютеров, входящих в домен Active Directory, могут определяться сразу на трех уровнях (на уровне локального компьютера, на уровне домена и на уровне организационной единицы), как сразу становится понятно, что порой разобраться в действительных настройках групповой политики для отдельного пользователя бывает довольно сложно. Именно для облегчения работы с подобными групповыми политиками и была разработана оснастка Результирующая политика. Она собирает настройки групповых политик конкретного пользователя сразу на всех трех уровнях Active Directory и выводит общую групповую политику, отображая при этом проблемные правила политики (которые конфликтуют на уровне реестра с другими правилами политики).

При этом в процессе поиска активных групповых политик оснастка использует базу данных WMI. Для получения сведений об ADM-файлах оснастка подключается даже к локальному компьютеру с использованием его сетевого имени (через стандартную скрытую общедоступную папку admin$). Оснастка Результирующая политика имеет GUID-номер {6DC3804B-7212-458D-ADB0-9A07E2AE1FA2}.

Оснастка Результирующая политика входит в стандартную консоль Windows XP gpedit.msc. В процессе запуска этой консоли (а также в процессе ее работы) для протоколирования ошибок используются файлы из каталога %systemroot%\Debug\UserMode, а также идет просмотр ветви реестра HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon. Для работы оснастки Результирующая политика используются следующие параметры DWORD-типа.

- RSoPLogging — определяет, будет ли выполняться протоколирование результирующих политик на данном компьютере. Если значение данного параметра равно 0, то протоколирование результирующей политики будет отключено. По умолчанию значение равно 1. Параметр может также находиться в ветви HKEY_LOCAL_NACHINE\Software\Policies\Microsoft\Windows\System.
- GroupPolicyMinTransferRate — определяет порог скорости подключения к компьютеру при использовании результирующих политик или обновлении групповых политик, преодоление которого будет говорить консоли управления Microsoft, что используется медленное подключение. По умолчанию значение данного параметра равно 500. Другими словами, если реальная скорость подключения к компьютеру будет меньше 500 Кбайт/с, то оно будет считаться медленным. Если значение данного параметра равно 0, то все подключения будут считаться быстрыми. Параметр может принимать значения от 0 до 0xFFFFFFA0. Параметр может также находиться в ветви реестра HKEY_LOCAL_NACHINE\SOFTWARE\Policies\Microsoft\Windows\System (а также в ветви корневого раздела HKEY_CURRENT_USER). По умолчанию параметры не существуют.

После открытия консоли отобразится окно, подобное окну консоли Групповая политика. Оно также содержит элементы Конфигурация компьютера и Конфигурация

пользователя с вложенными разделами. Но, в отличие от консоли Групповая политика, консоль Результирующая политика включает в себя лишь установленные или отключенные правила групповой политики — правила в состоянии Не задано в консоли Результирующая политика не отображаются. Еще одно отличие консоли Результирующая политика от консоли Групповая политика заключается в том, что первая позволяет лишь просмотреть правила, а не редактировать их.

Как уже говорилось ранее, Результирующая политика позволяет определить проблемные правила — напротив них отображается красный крестик. При этом на вкладке Информация об ошибке диалога Свойства таких вкладок можно определить проблему установки правила.

ПРИМЕЧАНИЕ

В диалоге Свойства можно также определить уровень, к которому принадлежит данное правило. Для этого предназначена вкладка Приоритет.

Шаблоны безопасности

Шаблоны безопасности являются довольно интересной оснасткой, с помощью которой можно создать свой собственный шаблон безопасности или отредактировать уже существующие. Шаблоны безопасности представляют собой файлы, хранящие различные настройки параметров реестра и файловой системы Windows XP. Файлы шаблонов безопасности можно легко импортировать в систему, чтобы настройки, в них содержащиеся, были применены к данному компьютеру.

ПРИМЕЧАНИЕ

Шаблоны безопасности нельзя применить к компьютерам, операционная система которых установлена на дисках, имеющих файловую систему, отличную от NTFS.

Оснастка Шаблоны безопасности не входит ни в одну стандартную консоль, поэтому для получения доступа к ней необходимо воспользоваться консолью управления Microsoft `mmc.exe`. Оснастка Шаблоны безопасности имеет GUID-номер `{5ADF5BF6-E452-11D1-945A-00C04FB984F9}`, поэтому если оснастка с таким номером будет запрещена с помощью групповых политик, то вы не сможете запустить Шаблоны безопасности (она просто исчезнет из списка доступных для открытия оснасток).

После открытия оснастки перед вами отобразится окно, подобное приведенному на рис. 11.10.

Оснастка Шаблоны безопасности по умолчанию содержит раздел `C:\WINDOWS\security\templates`. Этот раздел, в свою очередь, включает в себя набор стандартных шаблонов безопасности. Все отображаемые в оснастке шаблоны безо-

пасности находятся в каталоге файловой системы `C:\WINDOWS\security\templates`. При этом стоит сказать, что каталог, из которого берутся шаблоны безопасности, не статичен. Другими словами, путь к каталогу можно изменить с помощью реестра (соответственно изменится и название раздела элемента Шаблоны безопасности). Для изменения пути к каталогу шаблонов безопасности необходимо изменить название раздела `C:/WINDOWS/security/templates` из ветви реестра `HKEY_CURRENT_USER\Software\Microsoft\Windows NT\CurrentVersion\SecEdit\Template Locations`.

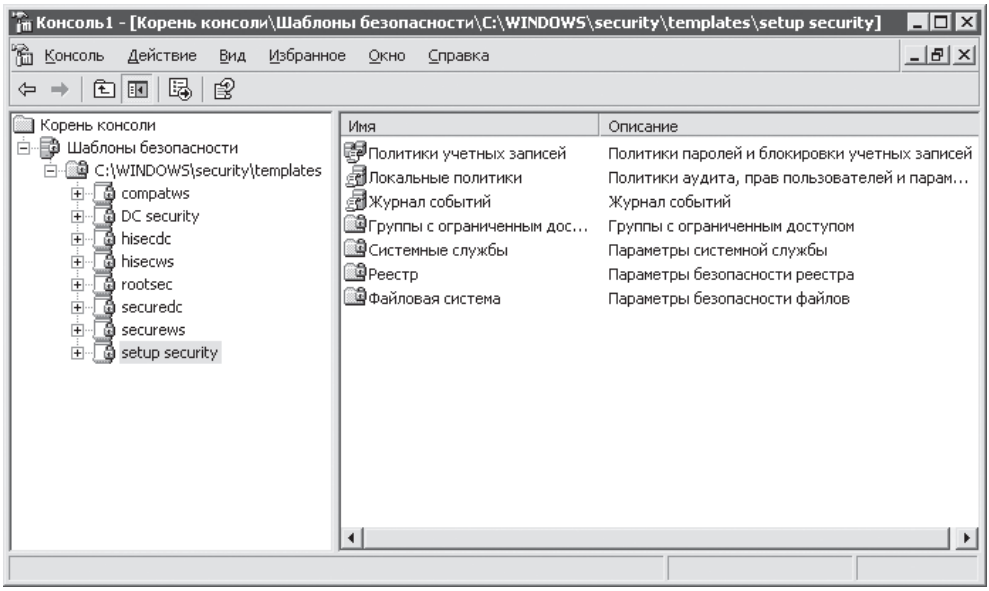


Рис. 11.10. Окно оснастки Шаблоны безопасности

ПРИМЕЧАНИЕ

Вы можете создать и свой собственный раздел в ветви реестра HKEY_CURRENT_USER\Software\Microsoft\Windows NT\CurrentVersion\SecEdit\Template Locations. Созданный вами раздел будет отображаться в оснастке наряду со стандартным разделом. Чтобы создать новый раздел с помощью механизмов оснастки, нужно выбрать в меню Действие команду Новый путь для поиска шаблонов.

Раздел C:/WINDOWS/security/templates ветви реестра HKEY_CURRENT_USER\Software\Microsoft\Windows NT\CurrentVersion\SecEdit\Template Locations может хранить параметр строкового типа Description, определяющий описание содержимого каталога. Это описание отображается в столбце Описание правой панели оснастки.

Все шаблоны безопасности, отображенные в оснастке, изменяют одни и те же параметры файловой системы и реестра (просто каждый шаблон устанавливает свои собственные значения этих параметров), поэтому сначала будут подробно рассмотрены изменяемые шаблонами параметры, а потом отличия в значениях этих

параметров для разных шаблонов безопасности. Для рассмотрения параметров воспользуемся шаблоном безопасности Setup security. Он используется сразу после установки операционной системы Windows XP для настройки доступа к файловой системе компьютера и ветвям реестра по умолчанию.

ПРИМЕЧАНИЕ

Шаблоны безопасности являются обычными файлами с расширением INF, расположенными в каталоге C:/WINDOWS/security/templates (по умолчанию). При этом название INF-файла используется в оснастке Шаблоны безопасности как название шаблона. Другими словами, шаблон Setup security является INF-файлом с именем Setup security.inf.

Содержимое шаблонов безопасности

Все шаблоны безопасности содержат следующие разделы: Политики учетных записей, Локальные политики, Журнал событий, Группы с ограниченным доступом, Системные службы, Реестр и Файловая система. Вкратце рассмотрим каждый из этих разделов.

Политики учетных записей

Раздел Политики учетных записей по умолчанию содержит три политики. Это Политика блокировки учетной записи, Политика паролей и Политика Kerberos.

- Политика паролей — с ее помощью можно настроить параметры создания паролей для учетных записей пользователей компьютера, а также определить параметры хранения паролей пользователей. Для этого применяются следующие правила.

ПРИМЕЧАНИЕ

Скорее всего, все приведенные ниже правила хранятся в ветви системного реестра HKEY_LOCAL_MACHINE\SECURITY.

  - Максимальный срок действия пароля — указывает количество дней, в течение которого будут действовать пароли пользователей. По истечении указанного срока пользователи должны сменить пароль. Для шаблона безопасности Setup security это правило равно 42 дням.
  - Минимальная длина пароля — определяет, из какого количества символов должен состоять (как минимум) создаваемый пароль, чтобы система разрешила его использование. Для шаблона безопасности Setup security это правило равно 0.
  - Минимальный срок действия пароля — указывает количество дней, которое должно истечь, чтобы пользователь смог сменить пароль. Если указанное

количество дней не истекло, то пользователю будет запрещено изменять пароль. Значение данной политики должно быть меньше значения политики Максимальный срок действия пароля. Для шаблона безопасности Setup security это правило равно 0.

- Пароль должен отвечать требованиям сложности — если данное правило установлено, то система не разрешит создание паролей, состоящих только из цифр или только из букв. При использовании данного правила все пароли должны содержать не меньше шести символов, находящихся в разных регистрах, а также не принадлежащих к алфавитно-цифровой клавиатуре (например, символы «&», «$», «!»). Для шаблона безопасности Setup security это правило отключено.
- Требовать неповторяемости паролей — значение данного правила определяет количество паролей, которые должны быть добавлены в базу данных SAM (содержит хэши паролей всех учетных записей пользователей), после чего система разрешит в качестве пароля задать уже использовавшийся ранее пароль. Для шаблона безопасности Setup security это правило равно 0 паролей.
- Хранить пароли всех пользователей в домене, используя обратимое шифрование — если данное правило будет включено, то система будет создавать пароли пользователей с возможностью их расшифровки (так называемое обратимое шифрование). Создание паролей с возможностью их расшифровки может потребоваться некоторым приложениям для аутентификации пользователя (например, это необходимо протоколу CHAP). Но перед установкой этого правила следует учесть, что такой способ хранения паролей резко снижает уровень безопасности компьютера. Для шаблона безопасности Setup security это правило отключено.

■ Политика блокировки учетной записи — с помощью данной политики можно определить правила поведения системы в случае нескольких попыток неудачного ввода пароля при аутентификации пользователя.

- Блокировка учетной записи на — определяет количество минут, на которое будет выполняться блокировка учетной записи после нескольких попыток неудачного ввода пароля. Значение может находиться в диапазоне от 1 до 99999 (если значение равно 0, то учетная запись будет заблокирована до тех пор, пока администратор компьютера ее не разблокирует самостоятельно). Для шаблона безопасности Setup security это правило не определено.
- Пороговое значение блокировки — указывает количество попыток неверного ввода пароля, после которых учетная запись будет заблокирована. Возможные значения лежат в пределах от 0 до 999. Для шаблона безопасности Setup security это 0 ошибок.
- Сброс счетчика блокировки через — определяет количество минут, по истечении которых счетчик неверных попыток ввода пароля будет обнулен.

Значение может находиться в пределах от 1 до 99999. Для шаблона безопасности Setup security это правило не определено.

- Политика Kerberos — определяет настройки протокола Kerberos, используемые при входе пользователя в систему. В контексте данной книги настройки данной политики рассмотрены не будут, так как они относятся к компьютерам, находящимся в домене, а это большая редкость на домашних компьютерах.

Локальные политики

Раздел Локальные политики содержит три политики: Политика аудита, Назначение прав пользователя и Параметры безопасности.

- Политика аудита — позволяет определить события, факты происхождения которых будут записываться в журнал Безопасность оснастки Просмотр событий. Можно указать запись в журнал Безопасность сведений об успешных или неудачных попытках выполнения следующих операций: вход в систему, доступ к объектам, имеющим собственный SACL (например, к принтерам, файлам, папкам), доступ к каталогам Active Directory и других. Для шаблона безопасности Setup security все события аудита, кроме аудита доступа к службе каталога (этот аудит не определен), отключены.
- Назначение прав пользователя — с помощью данной политики можно определить права различных пользователей или групп пользователей на выполнение различных операций с объектами и компонентами операционной системы. Например, с помощью этой политики можно определить пользователей, которым разрешено входить локально в систему, разрешено входить в систему через службу терминалов, разрешено выполнять архивирование файлов и каталогов и т. д.
- Параметры безопасности — с помощью данной политики можно настроить очень многие параметры реестра, относящиеся к безопасности компьютера. Довольно часто в Интернете можно прочитать советы об изменении тех или иных параметров реестра, настраивающих безопасность компьютера. Многие из параметров, указанных в таких советах, можно изменить и с помощью политики Параметры безопасности. Например, к наиболее часто упоминаемым в Интернете способам настройки безопасности с помощью реестра, которые также можно изменить и с помощью политики Параметры безопасности, относятся следующие.
  - Очистка файла подкачки pagefile.sys при завершении работы компьютера. Для шаблона Setup security данное правило отключено.
  - Сообщение, отображаемое перед входом пользователя в систему. Для шаблона Setup security данное правило не определено.
  - Посылать незашифрованный пароль сторонним SMB-серверам. Для шаблона Setup security данное правило отключено.
  - Запретить изменение паролей учетных записей пользователей. Для шаблона Setup security данное правило не определено.

- Пути в реестре, доступные через удаленное подключение. Для шаблона Setup security данное правило не определено.
- Разрешить анонимный доступ к общим ресурсам. Для шаблона Setup security данное правило не определено.
- Отключение или переименование учетных записей администратора и гостя. Для шаблона Setup security эти правила не определены (кроме отключения учетной записи гостя, по умолчанию эта запись отключена).

При этом большинство правил списка Параметры безопасности хранятся в реестре (то есть вы и сами можете добавить к данному списку свои правила изменения параметров реестра, чтобы изменять их с помощью шаблона безопасности). Для этого предназначена ветвь реестра HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\SeCEdit\Reg Values. Она содержит разделы, названия которых соответствуют пути к изменяемому правилом параметру реестра (данный путь должен начинаться не с корневого раздела ветви, а с класса, в котором хранится объект операционной системы (операционная система Windows XP является объектно-ориентированной), например класс Machine определяет корневой раздел HKEY_LOCAL_MACHINE). Эти разделы хранят следующие параметры.

- DisplayChoices — этот параметр строкового типа определяет описание возможного состояния правила (если для установки правила используется список состояний), а также значение, которое будет присваиваться параметру при установке соответствующего состояния правила.
- DisplayName — параметр строкового типа, определяет название правила, отображаемое в списке политики Параметры безопасности.
- DisplayType — этот параметр DWORD-типа определяет способ указания состояния правила. Параметр может принимать следующие значения:
  - 1 — отобразить счетчик для указания состояния правила;
  - 2 — поле для ввода значения;
  - 3 — раскрывающийся список (для выбора возможного состояния из списка);
  - 4 — список для выбора состояния;
  - 6 — два флажка, с помощью которых можно включить или отключить правило.
- ValueType — этот параметр DWORD-типа определяет тип изменяемого данным правилом параметра реестра. Возможные значения:
  - 1 — строковый тип параметра;
  - 3 — тип параметра REG_BINARY;
  - 4 — тип параметра REG_DWORD;
  - 7 — тип параметра REG_MULTI_SZ.

На рис. 11.11 можно видеть пример описания правила в реестре.

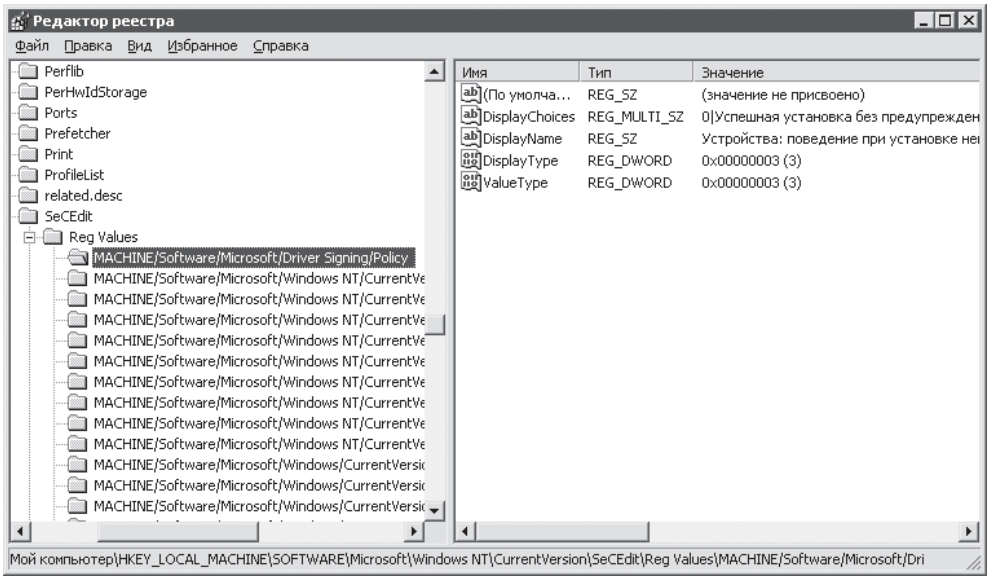


Рис. 11.11. Хранение правил политики Параметры безопасности

Журнал событий

С помощью данной политики можно настроить параметры стандартных журналов системы, доступ к которым можно получить с помощью оснастки Просмотр событий. Например, с помощью данной политики можно определить максимальные размеры файлов стандартных журналов системы, количество дней хранения этих файлов, а также запретить или разрешить просмотр системных журналов для учетной записи Гость. Все параметры реестра, изменяемые этой политикой, уже были рассмотрены в разделе о настройках оснастки Просмотр событий.

Группы с ограниченным доступом

С помощью данной политики можно добавить в группу временного пользователя (для повышения его прав на некоторое время). При этом после перезагрузки данный пользователь будет удален из группы. Тем самым администратор может делегировать на время права некоторым пользователям, не заботясь о снятии делегированных прав с пользователя — это выполнит система. Для добавления временного пользователя в группу нужно создать саму группу, в которую будет добавляться пользователь. Для этого нужно в контекстном меню раздела Группы с ограниченным доступом выбрать команду Добавить группу. После этого система предложит вам ввести или выбрать из списка группу, а затем предложит добавить в нее новых пользователей.

Системные службы

С помощью данной политики можно указать тип запуска служб, установленных на компьютере, или вообще отключить запуск некоторых служб.

Реестр

С помощью данного раздела можно указать права доступа к различным ветвям реестра. Чтобы указать права доступа к ветви реестра, необходимо сначала добавить в данный раздел ветвь реестра. Для этого необходимо в контекстном меню раздела Реестр выбрать команду Добавить раздел. После этого консоль управления Microsoft предложит вам указать права для доступа к данной ветви реестра.

Файловая система

С помощью этого раздела можно указать права доступа к различным каталогам файловой системы Windows XP. Чтобы указать права доступа к каталогу, необходимо сначала добавить в раздел Файловая система путь к каталогу. Для этого необходимо в контекстном меню раздела выбрать команду Добавить файл. После этого консоль управления Microsoft предложит вам указать права для доступа к данному каталогу или файлу, а затем определить, будут ли указанные вами права распространяться на все вложенные в каталог папки.

Стандартные шаблоны безопасности

Теперь рассмотрим другие стандартные шаблоны безопасности и их отличие от шаблона по умолчанию.

Compatws.inf

Настройки шаблона по умолчанию ограничивают группу Пользователи, запрещая ей доступ ко многим ветвям реестра и каталогам файловой системы. Вдобавок к группе Пользователи шаблон по умолчанию создает группу Опытные пользователи, обладающую большими правами в операционной системе. Создание двух разных групп необходимо для повышения безопасности. Тем не менее не рекомендуется использовать группу Опытные пользователи, так как она в системе имеет очень многие права, которые можно использовать не только на благо, но и во вред. В частности, пользователи из группы опытных имеют больше шансов осуществить различные методы взлома, направленные на получение прав администратора или системы.

Хотя в некоторых случаях без применения группы Опытные пользователи обойтись нельзя. В частности, когда пользователи компьютера должны иметь права на установку программ, не сертифицированных для Microsoft. Такие программы используют для своей установки ветви реестра и каталоги файловой системы, доступ к которым закрыт для обычных пользователей. Если необходимо, чтобы пользователи могли устанавливать такие программы, то желательно применять шаблон безопасности Compatws, а не группу Опытные пользователи. Данный шаблон разрабатывался для предоставления группе Пользователи специальных прав, которых достаточно для установки большинства программ. При этом другие права, доступные группе Опытные пользователи, группе Пользователи не передаются, то есть в остальном группа Пользователи остается ограниченной. При установке шаблона Compatws все члены группы Опытные пользователи удаляются из нее, и ей предоставляются следующие права на каталоги файловой системы.

- `%programfiles%` — чтение/запись/чтение и выполнение/список содержимого файлов/изменение.

- `%systemroot%\downloaded program files` — чтение/запись/чтение и выполнение/список содержимого файлов/изменение.
- `%systemroot%\temp` — чтение/запись/чтение и выполнение/список содержимого файлов/изменение.
- `%systemroot%\sysvol` — права не определены ни для одной из групп.

Шаблон безопасности Compatws также изменяет права на ветви реестра из корневого раздела HKEY_CLASSES_ROOT. Но в данном случае скорее происходит не повышение прав группы Пользователи на содержимое данной ветви, а понижение прав группы Опытные пользователи, чтобы они не могли выполнять запись в данный корневой раздел реестра.

Securews.inf

Настройки этого шаблона повышают общую безопасность рабочей станции. Существует также шаблон безопасности securedc, выполняющий аналогичные действия для контроллера домена. В контексте данной книги будет рассмотрен шаблон securewc, так как второй шаблон предназначен для компьютеров, находящихся в домене и являющихся контролерами доменов, что довольно редко на домашних компьютерах. Итак, основные отличия шаблона securewc от шаблона по умолчанию заключаются в следующем.

- Минимальная длина пароля 8 символов.
- Минимальный срок действия пароля 2 дня.
- Пароль должен отвечать требованиям безопасности.
- Требование неповторяемости 24 последних паролей.
- Блокировка учетной записи на 30 минут.
- Пороговое значение ошибок ввода пароля равно 5.
- Сброс счетчика блокировки через 30 минут.
- Выполняется аудит следующих событий: неправильный ввод пароля при входе в систему, аудит всех событий входа в систему и управления учетными записями, а также любое изменение политик и отказ системы в предоставлении привилегий учетной записи пользователя.
- Изменений в доступе к реестру и файловой системе нет. Изменения в параметрах безопасности не рассматриваются, хотя если кратко, то они в основном заключаются в отказе от протоколов аутентификации LM и NTLM.

Hisecws.inf

Данный шаблон определяет повышенный уровень безопасности рабочей станции. Как и предыдущие два шаблона, он имеет своего двойника, предназначенного для настройки повышенного уровня безопасности контроллера домена (Hisecdc). Основные отличия шаблона Hisecws от шаблона securewc заключаются в следующем.

- Блокировка учетной записи на 0 минут (то есть до ее явной разблокировки администратором).

- Аудит всех событий безопасности, кроме событий отслеживания процессов (отключен) и событий доступа к службе каталогов Active Directory (не определен).
- Изменений в доступе к реестру и файловой системе нет. Изменения параметров безопасности в основном заключаются в требовании подписывания передаваемых по сети данных.

Rootsec.inf

Данный шаблон безопасности служит лишь одной цели — присвоению прав доступа к системному диску по умолчанию. Именно этот шаблон применяется для настройки прав на доступ к системному диску при установке операционной системы.

Notssid.inf

Данный шаблон безопасности предназначен лишь для исключения доступа учетной записи Terminal Server к файловой системе и реестру Windows XP. Если сервер терминалов не используется, то можно применить данный шаблон безопасности для исключения SID сервера терминалов из прав доступа к объектам системы, хотя, как подчеркивает Microsoft, присутствие SID сервера терминалов никоим образом не влияет на безопасность компьютеров.

Создание и импортирование шаблона безопасности

Теперь для примера попробуем создать свой собственный шаблон безопасности. Как правило, для этого лучше воспользоваться одним из стандартных шаблонов, а не создавать шаблон с нуля.

Создание шаблона безопасности

Чтобы создать шаблон безопасности на основе любого другого шаблона, необходимо в контекстном меню шаблона выбрать команду Сохранить как. Затем консоль управления Microsoft предложит вам указать имя нового шаблона, после чего он отобразится в дереве оснастки Шаблоны безопасности. Существует также возможность копирования отдельных разделов шаблона в другой шаблон. Для этого необходимо в контекстном меню раздела эталонного шаблона выбрать команду Копировать. После этого в контекстном меню того же раздела, но шаблона-приемника нужно выбрать команду Вставить.

Например, чтобы быстро создать шаблон на основе шаблона Securews, но с настройками файловой системы из шаблона Rootsec, необходимо сначала создать шаблон на основе шаблона Securews (команда Сохранить как), а после этого скопировать раздел Rootsec ▸ Файловая система в раздел Файловая система созданного вами шаблона. После этого можно самостоятельно отредактировать состояние отдельных правил политик созданного вами шаблона.

ПРИМЕЧАНИЕ

Не рекомендуется изменять состояния правил непосредственно в стандартных шаблонах. Лучше для этого создать новый шаблон на основе одного из стандартных.

Импортирование шаблона безопасности

Шаблон безопасности создан. Но что теперь с ним делать? Для ответа на данный вопрос можно воспользоваться либо оснасткой Групповая политика, либо оснасткой Анализ и настройка безопасности. Можно также воспользоваться командой командной строки `secedit.exe`.

- Групповая политика — когда рассматривалась оснастка Групповая политика, были пропущены такие ее разделы, как Политики учетных записей и Локальные политики. Теперь вы знаете, что хранится в этих разделах, а также умеете создавать свои собственные шаблоны. Если вы уже создали свой шаблон с изменениями состояния правил данных разделов, то существует возможность его импортирования в групповую политику, чтобы настройки из шаблона применялись вместе с настройками групповой политики. Для этого необходимо в контекстном меню элемента Параметры безопасности консоли Групповая политика выбрать команду Импорт политики. После этого консоль управления Microsoft попросит указать путь к шаблону безопасности и использует его содержимое для настройки разделов Политики учетных записей и Локальные политики. При этом остальные настройки шаблона безопасности применяться не будут.
- Анализ и настройка безопасности — с помощью данной оснастки можно не только применить к компьютеру любой созданный шаблон (в отличие от Групповой политики оснастка использует все содержимое шаблона, а не только настройки разделов Политики учетных записей и Локальные политики), но и проанализировать текущие настройки компьютера с настройками из шаблона безопасности. Оснастка Анализ и настройка безопасности имеет GUID-номер `{011BE22D-E453-11D1-945A-00C04FB984F9}`. После ее добавления к консоли управления Microsoft в дереве консоли отобразится единственный элемент — Анализ и настройка безопасности. Если вы раньше никогда не использовали данную оснастку, то перед началом работы с ней необходимо создать базу данных текущих настроек безопасности компьютера. Для этого в контекстном меню оснастки необходимо выбрать команду Открыть базы данных и в отобразившемся диалоге ввести имя новой базы данных. После этого консоль управления Microsoft предложит указать имя шаблона безопасности, настройки которого будут импортированы в созданную базу данных (если вы используете уже существующую базу, то можно очистить ее содержимое перед импортом настроек шаблона безопасности).

ПРИМЕЧАНИЕ

Несмотря на то, что создаваемая база данных хранит информацию о настройках компьютера в неудобном для чтения виде, тем не менее злонамеренные пользователи смогут ее использовать для анализа текущих настроек безопасности компьютера.

После создания или открытия базы данных настроек шаблона в контекстном меню оснастки станут доступными команды Анализ компьютера и Настроить компьютер.

Если вы хотите только определить, соответствуют ли текущие настройки безопасности настройкам, содержащимся в открытой базе данных шаблона безопасности, то необходимо воспользоваться командой Анализ компьютера. После этого консоль управления Microsoft предложит вам указать путь к текстовому файлу, в который будет записан журнал процесса анализа компьютера. Затем в дереве оснастки отобразятся разделы, аналогичные разделам шаблонов безопасности. Эти разделы будут хранить описание текущего состояния (на вашем компьютере) установленных в шаблоне безопасности правил. Если текущие настройки состояния правила на вашем компьютере соответствуют настройкам из шаблона, то напротив правила будет установлен зеленый флажок. Если же настройки состояния правила в шаблоне безопасности отличаются от текущих настроек на вашем компьютере, то напротив соответствующего правила будет установлен красный крестик (рис. 11.12).

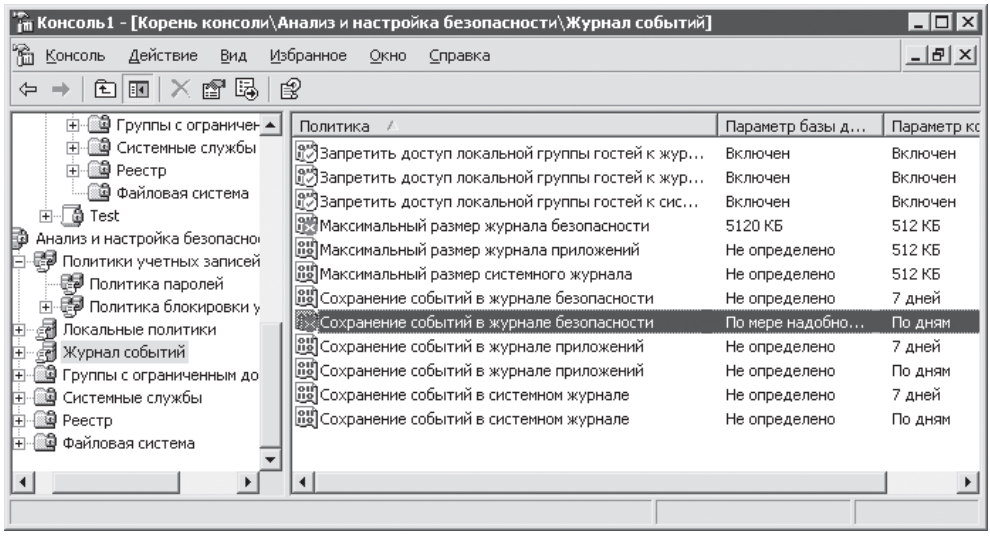


Рис. 11.12. Анализ текущей настройки безопасности компьютера

Чтобы установить настройки компьютера в соответствии с настройками из открытой базы данных, необходимо выбрать команду Настроить компьютер. После этого консоль управления Microsoft также предложит вам указать путь к текстовому файлу, используемому для хранения журнала процесса настройки компьютера.

С помощью команды `Secedit.exe` можно выполнить как настройку и создание шаблонов безопасности, так и анализ текущих настроек компьютера на основе шаблона безопасности или применение шаблона безопасности. Если работа с данной программой заинтересовала вас, то предлагаю воспользоваться стандартной справкой по данной программе, которую можно открыть с помощью команды `secedit.exe /?`.

Часть 4

Другие возможности Windows XP

Глава 12

Версии Windows

- **Статические параметры**
- **Динамические параметры**

Вот вы и прочитали значительную часть книги. Надеюсь, она вам понравилась и действительно оказалась полезной. Но перед тем, как ответить, поговорим еще о нескольких вопросах, которые не соответствуют общей теме книги, но все-таки могут быть вам интересны.

ПРИМЕЧАНИЕ

А общей темой книги, если по секрету, было обучение программированию с помощью сервера сценариев Windows и описание тех функций, которые можно будет использовать в своих сценариях.

Для начала поговорим о разнообразии версий Windows XP. Действительно, ведь существует не только две версии Windows — Professional и Home Edition. Есть также версия TabletPC для ноутбуков, 64-битная версия Windows, Windows .NET Server, Windows .NET Advanced Server, Windows .NET Datacenter Server. Существует также MediaCenterPC. Согласитесь, если бы Microsoft делала все эти версии отдельно, то Билл Гейтс давно бы обанкротился. Видимо, так же думали и программисты Microsoft, ведь информация о том, к какой версии Windows принадлежит ваша операционная система, заложена в самой операционной системе. И это нельзя однозначно назвать словом «плохо» или «хорошо» — такова жизнь, ведь это способ выживания большой корпорации. По этой причине не будем упоминать автора операционной системы Windows (ведь, несмотря на то, что Windows постоянно ругают, это действительно качественная и отличная операционная система), а лучше поговорим о том, где эта информация находится.

ВНИМАНИЕ

Ни автор, ни издательство «Питер», ни тем более корпорация Microsoft, не несут никакой ответственности за возможные последствия применения приведенных ниже трюков. Это незаконная операция, и ее описание приведено лишь в ознакомительных целях.

Статические параметры

Информация о текущей версии Windows находится в двух статических параметрах реестра и одном динамическом. Для начала поговорим о статических параметрах — они находятся в ветви реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\ProductOptions`. Это параметры `ProductType` и `ProductSuite`.

- `ProductType` — параметр `REG_SZ`-типа. Определяет само направление данной версии Windows: то ли она предназначена для рабочих компьютеров, то ли для серверов, то ли для контроллеров домена. В зависимости от назначения Windows параметр может принимать следующие значения:
  - `WinNT` — данная версия системы Windows является рабочей станцией (Windows XP Professional, Windows XP Home Edition);

- LanmanNT — версия Windows является контроллером домена;
- ServerNT — данная версия Windows является сервером.

ПРИМЕЧАНИЕ

Вот что пишет об этом параметре сама Microsoft: «Свойство ProductType представляет дополнительные сведения о компьютере. Возможны следующие значения: \n1 — Рабочая станция, \n2 — Контроллер домена, \n3 — Сервер».

- ProductSuite — параметр REG_MULTI_SZ-типа. Определяет дополнительную градацию Windows и может принимать следующие значения:
  - Blade — определяет версию Windows для Windows 2003 Server, поэтому в книге не рассматривается;
  - Personal — данная версия Windows принадлежит к линейке Home Editions;
  - DataCenter — определяет версию Windows для Windows 2003 Server, поэтому в книге не рассматривается;
  - EmbeddedNT — указывает разновидность Windows Embedded;
  - Terminal Server — определяет версию Windows для Windows 2003 Server, поэтому в книге не рассматривается;
  - Small Business(Restricted) — указывает версию Windows для Windows 2003 Server, поэтому в книге не рассматривается;
  - BackOffice — определяет версию Windows для Windows 2003 Server, поэтому в книге не рассматривается;
  - CommunicationServer — указывает версию Windows для Windows 2003 Server, поэтому в книге не рассматривается;
  - Enterprise — определяет версию Windows для Windows 2003 Server, поэтому в книге не рассматривается;
  - Small Business — указывает версию Windows для Windows 2003 Server, поэтому в книге не рассматривается;
  - ConcurrentLimit — определяет версию Windows для Windows 2003 Server, поэтому в книге не рассматривается.

ПРИМЕЧАНИЕ

Об этом параметре Microsoft пишет: «Свойство ProductSuite содержит информацию об установленных и лицензированных дополнениях к операционной системе».

Как видите, мир Windows разнообразен, а теперь подумаем, что можно сделать с приведенной здесь информацией. А сделать можно лишь одно — преобразовать одну версию Windows в другую, что в некоторых кругах еще называется форсажем. К сожалению, значение параметра ProductType Windows изменить не разрешит —

это грубое нарушение лицензионных прав на вашу версию операционной системы (именно такое сообщение выводит система на «синем экране смерти» после изменения значения параметра `ProductType`). Но с некоторыми поправками можно изменить значение параметра `ProductSuite`.

ВНИМАНИЕ

Данные сведения приведены лишь в ознакомительных целях. Любое изменение версии Windows является нарушением прав на данный продукт и может не только вывести вашу систему из строя, но и автоматически сделать вашу лицензионную версию в глазах корпорации Microsoft пиратской, после чего вы не сможете пользоваться такими функциями, как Windows Update.

Для примера преобразуем Windows XP Home Editions в Windows XP Professional. После этого вы получите такие новые функции, как возможность управления операционной системой с помощью `mstsc` (удаленное управление Рабочим столом), а также возможность установки операционной системы в качестве контроллера домена.

Итак, если посмотреть на описанные выше значения для параметра `ProductSuite`, то можно заметить, что для Windows XP Home Editions это значение должно быть равно `Personal`. Это действительно так, но какое значение параметра должно быть для Windows XP Professional, ведь в приведенном описании этого значения нет? Все дело в том, что для Windows XP Professional значение параметра `ProductSuite` должно отсутствовать.

Если вы уже попробовали изменить значение параметра `ProductSuite`, то, без сомнения, знаете, что операционная система не даст это сделать. Будет выведено сообщение о нарушении прав лицензионного продукта, после чего предыдущее значение вернется на свое место (оно вернется даже тогда, когда это значение явно не соответствует ни одному из приведенных выше значений). Поэтому простым способом редактирования параметра вы ничего не добьетесь.

Вспомним о ветви реестра `HKEY_LOCAL_MACHINE\SYSTEM` все, что было написано в предыдущих частях книги. Итак, ветвь `HKEY_LOCAL_MACHINE\SYSTEM` хранит все сведения о драйверах и службах, зарегистрированных в системе. Она также хранит наиболее важную информацию о конфигурации операционной системы — если содержимое ветви `HKEY_LOCAL_MACHINE\SYSTEM` будет повреждено, то с большой долей вероятности вы не сможете загрузить операционную систему. Но ведь ошибки могут происходить не только по вине пользователя, но и по вине сторонних программ или самой операционной системы — это уже очень большая группа риска, а по теории вероятности, чем больше факторов риска, тем больше вероятность, что непредвиденное событие все-таки произойдет. Именно поэтому программистами Microsoft для страховки было решено продублировать всю критически важную информацию в нескольких ветвях реестра — так появились ветви реестра `HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001`, `HKEY_LOCAL_MACHINE\SYSTEM\ControlSet002` и т. д. Каждая из этих ветвей хранит конфигурацию системы в разные моменты времени, а ветвь реестра `HKEY_LOCAL_MACHINE\`

SYSTEM\CurrentControlSet является лишь ссылкой на один из приведенных разделов реестра Windows.

Теперь можно поставить еще один вопрос — если запрещено изменять параметр в ветви реестра HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet, то будет ли также запрещено изменять значение того же параметра в ветвях реестра HKEY_LOCAL_MACHINE\SYSTEM\ControlSetNNN? Оказывается, что значения параметров данных ветвей совершенно не защищены от изменений и Windows позволяет редактировать любые параметры, даже те, которые запрещено редактировать в разделе CurrentControlSet.

Осталось вспомнить еще одно — как же Windows выбирает, какой из разделов ControlSetNNN нужно использовать при следующей загрузке компьютера. Для этого применяются значения параметров DWORD-типа из ветви системного реестра HKEY_LOCAL_MACHINE\SYSTEM\Select. Эта ветвь может содержать следующие параметры.

- Default — именно этот параметр определяет, какая копия раздела ControlSet будет загружена при нормальной загрузке системы. Например, если его значение равно 2, то при обычной загрузке системы раздел CurrentControlSet будет ссылкой на содержимое раздела ControlSet002.
- Current — определяет номер текущей копии раздела ControlSet, на которую ссылается раздел CurrentControlSet.
- LastKnownGood — указывает номер копии раздела ControlSet, которая будет использоваться для построения содержимого раздела CurrentControlSet при использовании команды меню альтернативной загрузки Загрузка последней удачной конфигурации.
- Failed — определяет раздел ControlSet, при предыдущей загрузке которого произошел какой-то сбой и загрузка была прервана.

Когда известно, какой из разделов ControlSet для какой загрузки предназначен, возникает еще один вопрос — а значения какого из них правильнее и лучше всего редактировать? Здесь автор может посоветовать лишь исходя из своего опыта — лучше всего редактировать значения параметров из раздела, ссылка на который указана в значении параметра LastKnownGood, а потом пробовать запустить систему с помощью команды альтернативного окна загрузки Загрузка последней удачной конфигурации. Есть большая доля вероятности, что после редактирования параметров ваша операционная система не загрузится, и тогда вы просто сможете загрузить ее в обычном режиме. Причем все дело в том, что в большинстве случаев операционная система не загружается уже после регистрации в ней пользователя — требует ввода нового активационного ключа. Из предыдущих глав книги вы знаете, что, как только в системе регистрируется пользователь, ветвь реестра ControlSet, с помощью которой была выполнена загрузка, считается корректной и ссылка на нее указывается в качестве значения параметра LastKnownGood. Другими словами, если вы использовали ветвь ControlSet, описываемую в параметре Default, то при таком стечении обстоятельств окажется поврежденной как текущая конфигурация ControlSet, так и конфигурация, на которую будет ссылаться значение параметра LastKnownGood.

Вот и все. Теперь вы можете попытаться изменить версию своей операционной системы, но хотелось бы еще раз напомнить, что это является незаконным предприятием и рассказ о нем приведен лишь в ознакомительных целях.

Динамические параметры

Теперь поговорим еще о нескольких параметрах, которые влияют на версию системы Windows. Автор назвал их динамическими, потому что в отличие от параметра `ProductSuite`, который был рассмотрен выше, эти параметры очень часто проверяются системой — особенно их любят проверять пакеты MSI. К тому же, предположительно, эти параметры работают только в Windows XP Professional — в других операционных системах применяются другие параметры.

Все приведенные ниже параметры находятся в ветви реестра `HKEY_LOCAL_MACHINE\SYSTEM\WPA`, хранящей все ключи активации, доступные на компьютере. Но, кроме данных ключей, эта ветвь системного реестра может включать в себя два раздела — `TabletPC` и `MediaCenter`. В каждом из них может присутствовать `DWORD`-параметр `Installed`, значение которого определяет, является ли текущая операционная система данной разновидностью Windows. Например, если значение этого параметра ветви реестра `TabletPC` будет равно `1`, то после перезагрузки вы сможете увидеть картину, подобную приведенной на рис. 12.1.

Рис. 12.1. Изменение версии Windows

ПРИМЕЧАНИЕ

Естественно, что лишь один из параметров Installed должен быть равен 1. Второй в таком случае либо не должен вообще существовать, либо должен быть равен 0.

Если вам понравился предыдущий рисунок и вы уже попытались изменить соответствующий вашим желаниям параметр, то возможны два варианта. Первый состоит в том, что в вашей системе не существует разделов `TabletPC` и `MediaCenter`. В этом случае вы безболезненно сможете изменить понравившийся вам параметр, создав его. Это был простой вариант. В сложном варианте в вашей системе уже существует раздел, в котором вы хотите создать или изменить значение параметра `Installed`. В этом случае система не разрешит вам этого сделать и единственный выход из такого положения — это воспользоваться второй операционной системой либо комплектом, подобным загрузочному диску ERD Commander, версия редактора которого способна изменять значения любых параметров.

Если у вас установлена вторая операционная система Windows, поддерживающая разделы NTFS, то можно загрузиться из нее, а потом в редакторе реестра просто загрузить куст `SYSTEM` первой операционной системы, параметр которой вы хотите изменить. Куст `SYSTEM` находится в ветви реестра `%systemroot%\system32\config` вашей первой операционной системы, и, чтобы его загрузить, необходимо выделить в редакторе реестра корневой раздел `HKEY_LOCAL_MACHINE`, а потом в меню Файл выбрать команду Загрузить куст. После этого перед вами появится диалог, в котором нужно выбрать куст `SYSTEM`, а в следующем диалоге присвоить ему произвольное имя — именно под таким именем вы сможете его увидеть как одну из ветвей корневого раздела `HKEY_LOCAL_MACHINE`.

ПРИМЕЧАНИЕ

Местоположение всех запущенных системой в данный момент кустов реестра можно определить по содержащимся в ветви реестра HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\hivelist параметрам строкового типа. Имена этих параметров определяют названия кустов, а значения — пути, по которым они расположены.

Если при преобразовании Windows XP Home Edition в Windows XP Professional ваша операционная система получит дополнительные возможности (подключение с помощью удаленного Рабочего стола и т. д.), то при преобразовании Windows XP в Windows XP TabletPC изменится только логотип Windows. Хотя если вы захотите, то можете установить дополнительные программы, предназначенные для TabletPC. Все они находятся в папке «`ïóòü ê äèñòðèáóòèâó îïåðàöèîííîé ñèñòåìû`»`\CMPNENTS\TABLETPC\I386` вашего установочного диска (конечно, вместо TabletPC ваш установочный диск может поддерживать только Windows XP MediaCenter, файлы которого находятся в каталоге «`ïóòü ê äèñòðèáóòèâó îïåðàöèîííîé ñèñòåìû`»`\cmpnents\mediactr\i386`). Все они

находятся в упакованном виде, поэтому для их извлечения понадобится выполнить команду `expand «ïóòü ê äèñòðèáóòèâó îïåðàöèîííîé ñèñòåìû»\ CMPNENTS\TABLETPC\ I386\*.* «ïóòü ê ïàïêå, â êîòîðóþ íóæíî ðàñïàêîâàòü ôàéëû (ïàïêà îáÿçàòåëüíî äîëæíà ñóùåñòâîâàòü)»`. После этого необходимо переименовать все расширения в соответствии с реальными расширениями, например, файл с расширением EX_ нужно переименовать в файл с расширением EXE.

ПРИМЕЧАНИЕ

Файлы установки дополнительных программ TabletPC и MediaCenter можно также найти в каталоге %systemroot%\INF. Они называются medctroc.INF и tabletpc.INF.

Напоследок хотелось бы напомнить, что изменение версии Windows — это незаконная процедура, которая может лишить вас лицензионных прав на вашу операционную систему и перевести ее в разряд пиратских. К тому же есть большая вероятность невозможности последующей загрузки. В частности, после изменения версии Windows XP на Windows XP MediaCenter с большой долей вероятности система попросит у вас новый активационный ключ или переустановку системы. А при изменении версии Windows XP на Windows Embedded существует большая вероятность того, что система вообще не загрузится, не выводя больше никаких сообщений.

Кстати, еще хотелось бы несколько слов сказать об установочном диске Windows XP. Вообще, установочные диски от Microsoft — это такая интересная вещь, в которой никогда точно не знаешь, что найдешь. Например, на установочном диске Windows XP Professional можно найти пакет установщика Windows для инсталляции программы MSN Messenger 6.1. Если вы часто пользуетесь программой Messenger (по умолчанию устанавливается Messenger версии 4.7), то можете попробовать воспользоваться MSN Messenger 6.1, вдруг понравится. Как и большинство файлов на установочном диске, пакет установки MSN Messenger 6.1 хранится в каталоге `i386` в сжатом виде. Пакет установки MSN Messenger 6.1 имеет название `MSNMSGS.MS_`. Чтобы его распаковать перед установкой, нужно воспользоваться командой в следующем формате:

```
expand "d:\MSNMSGS.MS_" "d:\1"
```

Здесь `d:\` определяет путь к каталогу, в котором расположен пакет установки MSN Messenger 6.1 (не забудьте перед распаковкой переместить пакет `MSNMSGS.MS_` в другой каталог, так как, скорее всего, непосредственно с установочного диска вам будет запрещено распаковывать файлы), а `d:\1` определяет каталог, в который будет распаковываться пакет установки. После распаковки пакета замените его расширение MS_ на MSI, а потом установите данный пакет установщика Windows. В результате у вас появится вот такая красивая программа (рис. 12.2).

Рис. 12.2. Окно программы MSN Messenger 6.1

Эту программу можно открыть с помощью файла `msnmsgr.exe`, расположенного в каталоге `%ProgramFiles%\MSN Messenger`.

Глава 13

Программа Debug

- **Команды программы**
- **Описание кода**
- **Простой пример**
- **Другие команды программы**

Следующей возможностью, которую хотелось бы описать, является способ создания программ Win32 с помощью стандартного отладчика командной строки `debug.exe`.

Но сначала хотелось бы пояснить, зачем вообще упоминается этот отладчик. Согласитесь, существуют компьютеры общего пользования, на которые пользователь не имеет права ничего устанавливать. На них нет привода компакт-дисков, дисковода, Интернета или нельзя скачивать из Интернета файлы. В общем, администраторы сделали так, чтобы нельзя было устанавливать свои файлы, в том числе вирусы, перехватчики клавиатуры, троянские кони. Но уже стало стандартом, что программа `debug.exe` входит в поставку Windows и о ней теперь знает все меньше и меньше администраторов...

Это был тонкий намек администраторам подобных общественных компьютеров.

ПРИМЕЧАНИЕ

Отладчик debug.exe входит в стандартную поставку всех версий Windows (находится в каталоге %systemroot%\system32) и предназначен для отладки DOS-приложений на языке «Ассемблер» или машинном языке.

Итак, запустим командную строку и введем название команды — `debug.exe`. После этого мы попадем в среду программы, о чем можно узнать из появившегося приветствия в виде горизонтального курсора.

Теперь посмотрим на сам код простенькой программы Win32, вызывающей стандартное окно приветствия.

ПРИМЕЧАНИЕ

Данный код был взят из статьи журнала «Ваш компьютер» за 2003 год.

13.1. Win32 Hello, World!

```
f 0 400 0
f 1000 1200 0
a 0
db 4d,5a

a 3c
db 40

a 40
db 50,45,0,0
dw  14c,1
```

```
a 54
dw e0,10f,10b

a 68
dw 10d0

a 74
dw 0,40,1000,0,200,0
dw 4,0,0,0,4

a 90
dw 2000,0,200,0,0,0,2

a b4
dw 10

a c0
dw 1090,0,3c

a 140
dw 1000,0,1000,0,200,0,200,0

a 15c
dw 20,e000

a 1010
db 54,68,69,73,20,54,69,74,6c,65,3f

a 1020
db 54,68,69,73,20,4d,65,73,73,61,67,65,3f

a 1040
db 55,53,45,52,33,32,2e,64,6c,6c

a 1050
db 4b,45,52,4e,45,4c,33,32,2e,64,6c,6c

a 1060
db 0,0,4d,65,73,73,61,67,65,42,6f,78,41
```

```
a 1070
db 0,0,45,78,69,74,50,72,6f,63,65,73,73

a 1080
dw 1060,0,0,0,1070,0,0,0

a 1000
dw 1060,0,0,0,1070,0,0,0

a 1090
dw 1080,0,0,0,0,0,1040,0,1000,0
dw 1088,0,0,0,0,0,1050,0,1008,0

a 10d0
db 6a,24
db 68,10,10,40,0
db 68,20,10,40,0
db 6a,0
db ff,15,0,10,40,0
db 6a,0
db ff,15,8,10,40,0

m 1000 1200 200
m 0 400 100
n«□□□□□□□□□□□□□□□□□□□□□□□□□bin»
r cx
400
w
q
```

Вот и весь код программы, написанной на машинном языке и вызывающей окно, отображенное на рис. 13.1.

Рис. 13.1 Результат выполнения программы

Команды программы

Теперь попробуем разобраться в этом коде. Для начала опишем команды отладчика, применяемые в нем.

F«□□□□□□□□□□□□□□»«□□□□□□□□□□□□□□»«□□□□□□□□□□□□»

Данная команда заполняет «заполнителем» диапазон памяти, начиная с «начального адреса» и заканчивая «конечным адресом».

A«□□□□□»

Команда говорит отладчику о том, что вы хотите изменить содержимое, записанное в памяти, начиная с указанного «адреса». После ее ввода перед вами появится приглашение, указывающее, какой участок памяти в данный момент редактируется. Чтобы сказать отладчику, что вы уже отредактировали необходимый вам участок памяти, нужно в пустой строке нажать клавишу Enter.

Можно также ввести данную команду без значения адреса — в этом случае вы будете редактировать значение адреса памяти, применяемого при последней операции останова.

m«□□□□□□□□□□□□□□»«□□□□□□□□□□□□□□»«□□□□□□□□□□□□□□»

Эта команда копирует содержимое диапазона памяти, начиная с «начального адреса» и заканчивая «конечным адресом», в область памяти, начинающуюся с адреса назначения.

N«□□□□□□□□□□□□»

Данная команда указывает, как будет называться создаваемый отладчиком файл (и в каком каталоге он будет находиться). Следует учитывать, что отладчик создавался для приложений MS-DOS, поэтому он не может создавать EXE-файлы — именно поэтому в коде и создается BIN-файл (после его создания нужно будет переименовать расширение BIN в расширение EXE).

Команда применяется также для загрузки файла и указания параметров запуска файла (в этом случае после команды `n` должны идти аргументы программы).

R«□□□□□□□»

Эта команда говорит отладчику о том, что вы хотите изменить содержимое конкретного регистра процессора. В контексте приведенного кода изменяется содержимое регистра `CX`. После ввода команды появится приглашение (в виде двоеточия) для ввода нового содержимого регистра.

Если ввести команду без указания конкретного регистра, то перед вами отобразится содержимое всех регистров процессора, всех флагов (определяют, было ли зарегистрировано переполнение при выполнении операции с числами, является ли число четным и т. д.) и содержимое данной области памяти.

Можно также отредактировать установки флагов. Для этого нужно ввести такую разновидность команды: `r f`, после чего перед вами отобразится установка флагов в текущий момент и приглашение для редактирования флагов. Чтобы отредактировать один из флагов, нужно ввести в приглашении противоположный ему флаг. Например, для флага `cy` (перенос) нужно указать флаг `nc` (нет переноса).

W«□□□□»

И наконец, с помощью этой команды записывается содержимое памяти на диск в виде программы Win32. Без аргумента данная команда начинает запись файла из адреса памяти CS:100, но можно самому указать адрес памяти, из которого будет начинаться запись.

Q

Эта команда закрывает окно отладчика.

Описание кода

Теперь, когда вы знаете описание необходимых команд, можно заняться описанием самого кода программы. И описывать его будем так: сначала указывается адрес памяти (или команда), а потом кратко говорится о том, для чего мы записываем по этому адресу памяти данные.

ПРИМЕЧАНИЕ

Еще перед описанием кода стоит сказать о командах db и dw, с которых начинается запись значений в адреса памяти — эти команды указывают на размер одной записываемой ячейки (ячейки отделяются запятыми). Если указана первая команда, то одна ячейка будет занимать в памяти 1 байт, а если указана вторая команда, то одна ячейка будет занимать 2 байта.

И еще одно — все значения в коде приведены в шестнадцатеричном виде и пишутся в обратном порядке.

Сначала нужно заполнить нулями (то есть очистить) диапазоны памяти от 0 до 400 и от 1000 до 1200. В первом диапазоне будет содержаться сама программа, а второй диапазон будет рабочим — именно в нем для удобства и будет вначале собрана программа.

Начиная с адреса `0` и заканчивая адресом `15ñ`, формируется заголовок PE-файла: вначале пишется заголовок DOS-файла (адрес `0`, записывается `MZ`), потом указывается, с какого адреса будет начинаться заголовок PE-файла (содержимое адреса `3c`), и в этом адресе пишется сам заголовок PE-файла (адрес `40`, записывается «Ð»,«Å»,`0,0`). По этому же адресу записывается идентификатор процессора, для которого предназначена программа (для i386 вводится `14ñ`), и количество секций, из которых она будет состоять. Дальше, по адресу `54`, указывается размер

NT-заголовка, флаги программы и «магическое значение». Если значения предыдущих адресов были статичны, то содержимое адреса 68 зависит от самой программы — оно указывает на адрес точки входа в программу. Начиная с этого адреса, будет вводиться сам код программы. По адресу 74 вводится базовый адрес загрузки (0,40), выравнивание в памяти и в файле (1000,0,200,0), а также версия операционной системы и версия подсистемы (4,0,0,0,4). По адресу 90 указывается размер образа с заголовками в памяти (2000), размер заголовка в файле (200) и подсистема (2). По адресу b4 указывается количество входов в каталоге смещений (10), а по адресу c0 описываются сами входы в каталог (мы используем только один): адрес таблицы импорта (1090) и ее размер (3ñ). По адресу 140 начинается таблица объектов (опять имеет один вход): занимаемый объем памяти (1000), с какого адреса начинается (1000), сколько места занимает в файле (200) и по какому смещению в нем находится (200). И последний адрес заголовка — 15ñ. В нем хранятся флаги (секция кодовая, имеет разрешения на чтение, запись и исполнение).

После формирования заголовка формируются данные программы — в диапазоне адресов от 1010 до 1070. Сначала записывается заголовок сообщения и его текст (адреса 1010 и 1020), потом названия библиотек, из которых будут взяты функции MessageBox (выводит наше окно) и ExitProcess (завершает программу) (адрес 1040 — USER32.DLL, а адрес 1050 — KERNEL32.DLL). И наконец, сами названия функций (адрес 1060 — MessageBoxA и адрес 1070 — ExitProcess), которые пишутся с учетом регистра.

Теперь нужно сформировать таблицу поиска, импортируемых адресов и таблицу импорта.

Таблица поиска содержит адреса функций и способ их поиска в библиотеках (по порядковым номерам или именам). В нашем случае будем вести поиск по именам, поэтому таблица поиска, расположенная по адресу 1080, принимает такой вид: 1060,0,0,0,1070,0,0,0. Здесь 1060,0 указывает на функцию MessageBoxA (0 отделяет названия функций между собой), потом идет 0,0 — разграничитель между функциями различных библиотек, а 1070,0 — адрес функции ExitProcess из другой библиотеки. Следует также учитывать, что первые два байта перед названиями функций должны быть равны 0 — они являются индексом, по которому в библиотеке должны находиться функции, но поскольку индекс неизвестен, нужно оставить эти поля пустыми, чтобы загрузчик сам нашел в библиотеках данные функции.

Таблица импортируемых адресов располагается в самом начале секции (в нашем случае по адресу 1000) и содержит адреса импортируемых функций — аналог таблицы поиска.

Таблица импорта связывает библиотеки с таблицами поиска и импортируемых адресов (в коде начинается по адресу 1090). Каждая строка таблицы описывает одну библиотеку (сначала содержится адрес начала описания функций из библиотеки в таблице поиска (1080,0), потом два пустых поля (0,0,0,0), потом адрес, хранящий название библиотеки (1040,0), и адрес начала описания функций данной

библиотеки в таблице импортируемых адресов (1000,0)). Аналогично описывается библиотека KERNEL32.DLL. После описания всех библиотек нужно оставить еще одну пустую строку таблицы импорта, то есть следующие 20 байт. Итого размер таблицы импорта равен $3 \times 5 \times 4 = 60$, а в шестнадцатеричном виде — 3c, что мы и вводили в заголовке PE по адресу c0.

И наконец, последняя часть кода — сам код. Он начинается с адреса 10d0, который и является точкой входа в программу — ее мы и указывали в заголовке PE файла по адресу 68.

Код довольно прост, но написан на машинном языке:

```
db 6a,24
db 68,10,10,40,0
db 68,20,10,40,0
db 6a,0
db ff,15,0,10,40,0
db 6a,0
db ff,15,8,10,40,0
```

Так вызывается функция MessageBoxA и ей передаются необходимые параметры: сначала помещается значение 24 (указывает, что вызываемое окно имеет две кнопки и значок вопроса), потом адрес заголовка, адрес сообщения и 0 (дескриптор родительского объекта, которого у нас нет). Если описать приведенный код более просто, то получится:

- Push 24 — поместить в стек значение 24;
- Push offset «ïåðåìåííàÿ ñ çàãîëîâêîì îêíà» — поместить в стек адрес памяти, содержащий заголовок окна;
- Push offset «ïåðåìåííàÿ ñ ñîîáùåíèåì îêíà» — поместить в стек адрес памяти, хранящий сообщение окна;
- Push 0 — поместить в стек 0;
- Call «àäðåñ ïàìÿòè, ñîäåðæàùèé íàçâàíèå ôóíêöèè».

Аналогично вызывается функция ExitProcess.

Вот и все. Теперь только осталось скопировать диапазон адресов от 1000 до 1200, хранящий нашу программу, в память, начиная с адреса 200, а потом сместить всю программу на 100, так как при сохранении файла отладчик обрезает первые 100h байт памяти.

Простой пример

Конечно, приведенный выше пример довольно сложен — ведь он написан на машинном языке и в шестнадцатеричном виде. Но его можно упростить, ведь отладчик в Windows XP поддерживает как ASCII-символы, так и язык «Ассемблера».

Вот упрощением мы сейчас и займемся. Например, напишем программу, которая будет открывать созданный нами ранее файл (если он будет называться `hello.exe` и находиться на диске `D:`).

13.2. [illegible]

[illegible]

```
a 1010
db "D:\HELLO.EXE"

a 1040
db  "SHELL32.DLL"

a 1050
db  "KERNEL32.DLL"

a 1060
db 0,0,"ShellExecuteA"

a 1070
db 0,0,"ExitProcess"

a 1080
dw 1060,0,0,0,1070,0,0,0

a 1000
dw 1060,0,0,0,1070,0,0,0

a 1090
dw 1080,0,0,0,0,0,1040,0,1000,0
dw 1088,0,0,0,0,0,1050,0,1008,0

a 10d0
xor bx,bx
push bx
push bx
push bx
db 68,10,10,40,0
push bx
push bx
db ff,15,0,10,40,0
```

```
push bx
db ff,15,8,10,40,0

m 1000 1200 200
m 0 400 100
n d:\rr.bin
r cx
400
w
```

Согласитесь, уже проще — ведь теперь не нужно вводить текстовую информацию в шестнадцатеричном виде. А если учесть еще одну возможность командной строки, то станет совершенно просто. Все дело в том, что совсем не обязательно вводить данный текст в отладчике, ведь можно воспользоваться Блокнотом, а потом просто скопировать введенный текст в буфер обмена, запустить `debug.exe` и нажать правую кнопку мыши в области его окна, чтобы отладчик начал обработку команд из буфера обмена. При этом только следует убедиться, что режим быстрой вставки в командной строке включен (`DWORD`-параметр `QuickEdit`, расположенный в ветви реестра `HKEY_CURRENT_USER\Console`, должен быть равен `1`).

Другие команды программы

Программа `debug.exe` содержит и другие команды, но в контексте данной книги они описаны не будут. Если вас заинтересовала приведенная информация, то для начала предлагаю посмотреть описание команд программы `debug.exe` в Центре справки и поддержки операционной системы Windows XP.

Глава 14

Безопасность

- **Угроза получения учетной записи администратора с помощью учетной записи опытного пользователя**
- **Системная учетная запись**

В данной главе рассмотрим некоторые вопросы безопасности функционирования системы. В частности, опишем недокументированную угрозу получения учетной записи администратора с помощью учетной записи опытного пользователя и рассмотрим вопросы использования системной учетной записи.

Угроза получения учетной записи администратора с помощью учетной записи опытного пользователя

Как говорилось раньше, использование группы Опытные пользователи не приветствуется Microsoft, так как данная группа имеет очень многие права в системе. В этой главе хотелось бы рассказать еще об одной причине, по которой лучше не использовать группу Опытные пользователи. А именно, о способе получения прав администратора или отдельной его учетной записи в том случае, если на взламываемой машине у вас есть учетная запись опытного пользователя либо любая другая учетная запись, имеющая права на запуск служб. Например, этим способом может воспользоваться администратор, которому вы запретили создавать других администраторов, чтобы обойти ваше ограничение. Для работы данного метода на взламываемой системе также должны быть установлены дополнительные сервисы.

Для реализации данного метода взлома необходима сторонняя служба, файл которой в Windows не защищается WFP, а также к которой вы имеете доступ. Еще одним требованием к службе является обязательное разрешение взаимодействия с Рабочим столом, а значит, служба должна запускаться от имени системы. Как правило, такие службы появляются при установке антивирусов (например, «Антивируса Касперского»), различных пакетов для программирования (например, Microsoft Visual C++ или Microsoft Visual Studio .NET), программ получения сведений о системе (например, SySoftware Sandra), программ эмуляторов операционной системы (например, VMware) и других программ, которые требуют для своих нужд права администратора (рис. 14.1).

Если все условия выполняются, а, как правило, они выполняются уже потому, что сейчас офис или отдельный компьютер невозможен без установки антивируса, не говоря уже о других программах, то можно начать попытку взлома. Для этого нужно скопировать файл `explorer.exe`, расположенный в каталоге `%systemroot%`, в каталог, который используется для хранения файла службы, например, как показано на рис. 14.1, в каталог `%programfiles%\common files\microsoft shared\VS7Debug`. После этого необходимо переместить куда-нибудь файл службы (на рис. 14.1 это файл `mdm.exe`), а файлу `explorer.exe` присвоить имя перенесенного нами файла службы (в нашем случае `mdm.exe`).

Вот и все приготовления, которые могут занять у вас несколько минут. Дальше есть две линии сюжета: если служба на данный момент не запущена, то просто запустите ее и переходите к следующему абзацу книги. Если же служба в данный момент работает, то нужно перезагрузить компьютер (обязательно перезагрузить, если вы просто смените сеанс, то ничего не произойдет — служба не будет

перезапущена). После перезагрузки компьютера, еще до загрузки оболочки, но уже после ввода регистрационных данных пользователя, перед вами отобразится окно Проводника. Это говорит о том, что запустилась измененная служба (конечно, это произойдет только в том случае, если параметр `Type` для службы равен `110`, а параметр `Start` равен `2`, это означает, что служба является приложением и запускается при входе пользователя в систему сервисом `smss.exe`). На данный момент окно Проводника вам не нужно, поэтому просто закройте его. Теперь в Диспетчере задач (нажмите комбинацию клавиш Ctrl+Alt+Delete для его вывода) нужно самому запустить оболочку `explorer.exe`, так как ваша оболочка теперь самостоятельно не загрузится. Для этого в меню Файл выберите пункт Новая задача (Выполнить) и в появившемся диалоговом окне введите команду `explorer.exe`. Все, теперь вы полностью вошли в систему и при этом измененная вами сторонняя служба не запущена (все приведенные действия нужны потому, что опытные пользователи могут запускать службы, но не останавливать их). Осталось только одно — запустить оснастку `services.msc` и самостоятельно запустить измененную службу.

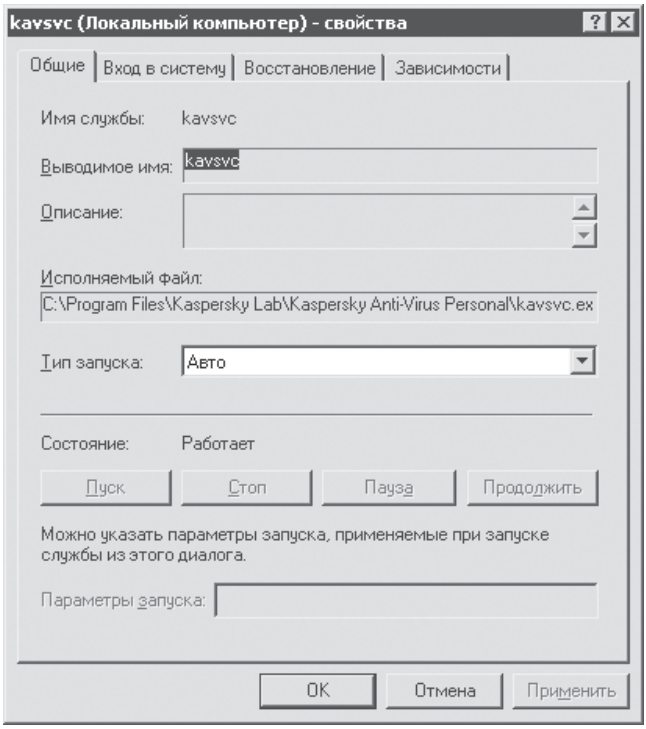


Рис. 14.1. Вот пример службы, файл которой не защищен WFP, так как он не находится в каталоге %systemroot%\system32

После запуска службы перед вами появится окно Проводника, в адресной строке которого нужно быстро ввести команду `%systemroot%\system32\lusrmgr.msc` (лучше просто поместить команду в буфер обмена перед выполнением данного шага взлома).

ПРИМЕЧАНИЕ

Естественно, вы можете ввести и другие команды, например команду %systemroot%\system32\cmd.exe, чтобы открыть окно командного процессора с правами системы (или учетной записи, от имени которой запускалась сторонняя служба).

Вы спросите, почему нужно вводить быстро? Все дело в том, что файл Проводника, которым вы заменили файл службы, не предназначен для запуска в качестве службы. Это значит, что он не ответит на запросы оснастки `services.msc` (хотя, как видите, это не мешает системе запустить файл оболочки). А если служба не отвечает на запросы, то через некоторое время система автоматически завершит такую службу.

Вот и все, теперь перед вами должна отобразиться соответствующая оснастка, которая будет запущена от имени системы (или учетной записи, от которой запускалась измененяемая вами служба). После отображения данной оснастки закройте окно Проводника. Конечно, система может и сама его закрыть по истечении времени ожидания ответа от службы, но лучше все-таки этого не ждать.

Как видите (рис. 14.2), система завершает неотвечающую службу, но не завершает другие процессы, порожденные завершенной службой, поэтому теперь вы можете спокойно редактировать группу, к которой принадлежит ваша учетная запись, либо создавать новую учетную запись администратора.

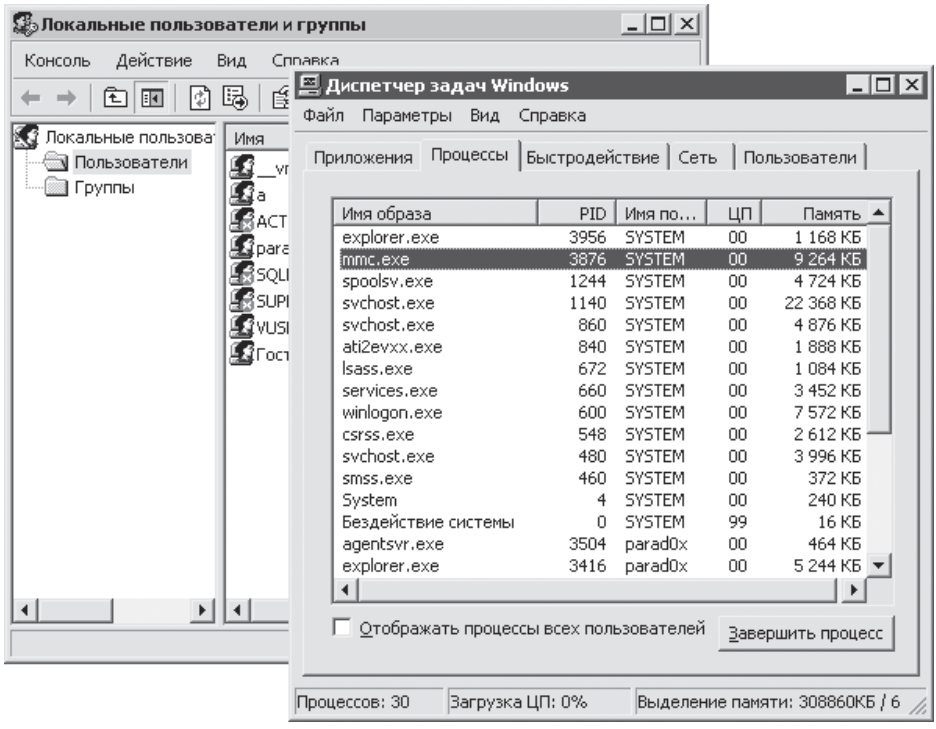


Рис. 14.2. Система завершила неотвечающую службу, но не завершила другие процессы, порожденные ею, поэтому у вас теперь есть оснастка с правами системы

Системная учетная запись

Как можно заметить из приведенного выше способа взлома, стандартной системной учетной записью можно воспользоваться для получения прав администратора. Это происходит потому, что такая запись имеет в чем-то даже больше прав, чем учетная запись администратора, и, как правило, администраторы компьютера не ограничивают учетную запись системы (а в основном просто не замечают угрозы этой записи). При рассмотрении групповых политик вы узнали, что настройки интерфейса Internet Explorer также записываются в ветвь реестра от имени учетной записи системы (от имени процесса `WINLOGON.EXE`). Это тоже довольно сложно понять. Если групповые политики может редактировать только администратор, то почему необходимо использовать процесс `WINLOGON.EXE`? Здесь также, кстати, скрывается возможность взлома — получение прав администратора или изменение любых элементов реестра и файловой системы компьютера с помощью процесса `WINLOGON.EXE`, запущенного от имени системы. Все дело в возможности импортирования настроек конфигурации браузера, программ на вкладке Программы или настроек ограничений в INF-файлах каталога `%systemroot%\system32\GroupPolicy\User\Microsoft\IEAK\BRANDING` с помощью элемента групповой политики Настройка Internet Explorer. Ведь если добавить в данные INF-файлы свои действия, например редактирование ветвей реестра или добавление/удаление объектов файловой системы, то при следующем доступе к политике Настройка Internet Explorer все добавленные вами строки будут выполнены от имени системы.

Единственным, что по умолчанию делает невозможным выполнение данного метода взлома, является запрет на работу с консолью Групповая политика, а также редактирование импортируемых INF-файлов для пользователей с учетными записями, отличными от группы Администраторы. Именно поэтому категорически запрещено изменять настройки доступа к каталогу `%systemroot%\system32\GroupPolicy\User` и его содержимому. Хотя и это только вершина айсберга, ведь неизвестно, как процесс `WINLOGON.EXE` создает INF-файлы. Может быть, можно подобрать такое значение параметра реестра (который он записывает в INF-файл), которое будет вызывать неправильную запись в INF-файл значения параметра реестра? Например, чтобы на основе значения параметра реестра в INF-файле создавалось не только значение этого параметра, но и новая строка, выполняющая произвольный код? Например, после некоторых экспериментов получилось создать в INF-файле «мусор», то есть произвольный текст после значения параметра, который не обрабатывается. А можно ли как-нибудь записать в одну строчку INF-файла две команды? Или указать в значении параметра реестра какой-либо специальный символ (ведь реестр поддерживает Unicode), который при обработке процессом `WINLOGON.EXE` или INF-файлом будет интерпретироваться как переход на другую строку? Надеюсь, программисты Microsoft могут с уверенностью ответить отрицательно на все эти вопросы, иначе перед нами очередной потенциальный способ выполнения любых операций с реестром и файловой системой Windows XP от имени системы. Причем этим способом смогут воспользоваться представители не только группы Опытные пользователи, но и группы Пользователи.

Другим интересным вопросом является озвучивание системных событий. Как оказывается, озвучивание события также выполняется процессом WINLOGON.EXE. При этом путь к музыкальному файлу для озвучивания события хранится в ветви реестра, доступной для редактирования любым пользователям. Здесь также возникают вопросы. А нельзя ли вместо озвучивания музыкального файла выполнить какой-нибудь произвольный код? Или вместе с музыкальным файлом? Или, например, указать путь к ярлыку музыкального файла, а этот ярлык, в свою очередь, будет ссылаться на музыкальный файл, доступ к которому вам как пользователю был запрещен. Да здесь, собственно, и ярлык не нужен, как оказывается, так можно прослушать даже тот файл, доступ к которому вам был полностью запрещен, но вы точно знаете, где этот файл находится. Как это сделать? Все музыкальные файлы для озвучивания событий описаны в ветви реестра HKEY_CURRENT_USER\AppEvents\Schemes\Apps. Например, параметр (Ïî óìîë÷àíèþ) ветви реестра HKEY_CURRENT_USER\AppEvents\Schemes\Apps\.Default\SystemHand\.Current определяет путь к файлу, который будет проигрываться процессом WINLOGON.EXE при возникновении критической ошибки. Другими словами, вы можете указать здесь путь к любому музыкальному файлу, а потом, например, просто ввести в диалоге Запуск программы любую строку, не ассоциированную с программой, например символ b. Скорее всего, система не найдет программы с названием b.exe, а значит, произойдет событие критической ошибки и будет воспроизведен необходимый вам файл.

Конечно, прослушивать запрещенные файлы смешно (если только на них не записана конфиденциальная информация), но что, если поискать системную службу, которая откроет для вас любой файл? Или, например, запишет свои данные в файл, путь к которому вы укажете? Например, в файл, доступ к которому вам запрещен, но который вы хотите повредить.

Именно по приведенным выше причинам я и опасаюсь программ и служб, запущенных от имени системы. Ведь даже сама Microsoft говорит о том, что нужно использовать как можно меньше учетных записей с административными правами, но почему-то забывает о системной учетной записи, которая также имеет административные права. А ведь сейчас каждая мало-мальски нужная служба, установленная на компьютере, хочет работать с правами системы, даже если эти права ей совершенно не нужны. При этом неизвестно, насколько эта служба устойчива к взлому. А по теории вероятности, чем больше в системе таких служб, тем больше шансов на взлом такой системы.

Интересна также сама необходимость учетной записи системы с полными правами. Например, зачем службам такие возможности, как изменение ACL объектов или создание учетных записей администраторов? Зачем тому же WINLOGON.EXE такие возможности? Намного безопасней будет создать несколько ограниченных учетных записей системы, выполняющих только определенные задачи, которые могут понадобиться той или иной службе.

Глава 15

INF-файлы

- **Основные сведения**
- **Дополнительные возможности**

Хотелось бы в этой главе описать некоторые возможности INF-файлов. Здесь не будет полностью рассмотрен язык INF-файлов и способы написания на нем сценариев, но тем не менее попробуем понять, как с помощью INF-файлов можно выполнять копирование и удаление файлов, создание и удаление параметров реестра, а также рассмотрим некоторые интересные возможности INF-файлов.

Основные сведения

INF-файлы предназначены для описания начального процесса установки новой программы или оборудования. Каждый INF-файл должен начинаться с заголовка. Этот заголовок определяет версию INF-файла, а также версию операционной системы, для которой этот INF-файл написан. От версии информационного файла (INF-файла) зависят те возможности, которые он поддерживает. Существует две версии INF-файлов — обычные и расширенные. В главе 1 уже рассматривались способы вызова обычных и расширенных INF-файлов. При этом расширенные INF-файлы поддерживают следующие новые возможности (это не все возможности, только основные): выполнение различных программ до или после выполнения INF-файла, архивирование изменяемых значений параметров реестра, а также вывод сообщений перед выполнением INF-файлов или после него.

ПРИМЕЧАНИЕ

Необходимость изучения INF-файлов во многом оправдана. Конечно, сейчас INF-файлы заменили пакетами установщика Windows и другими способами описания начальной установки программ. Тем не менее они обладают несколькими интересными возможностями, которые будут рассмотрены далее и которые довольно трудно выполнить без использования INF-файлов. С помощью INF-файлов можно также работать с реестром, даже когда возможность работы с программой regedit.exe и REG-файлами была отключена с помощью DWORD-параметра DisableRegistryTools, расположенного в ветви реестра HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Policies\system. И не следует забывать, что INF-файлы могут использоваться системными процессами, то есть на их основе можно выполнить взлом операционной системы.

Обычные INF-файлы

Возможности обычных INF-файлов поддерживаются и расширенными, поэтому эти возможности будут рассмотрены первыми. Обычные INF-файлы начинаются со следующего заголовка:

```
[Version]
Signature="$WINDOWS  NT$"
```

При этом после ключевого слова `Signature` идет описание версии операционной системы, которая будет поддерживать такие INF-файлы. Если после этого ключевого слова идет слово `$WINDOWS NT$`, то данный файл написан для операционных

систем семейства NT и работать с ним в операционных системах семейства Windows 9x нельзя. Если же после ключевого слова идет слово `$CHICAGO$`, то данный INF-файл был написан для операционных систем семейства Windows 9x. При этом работать с этим файлом можно будет и в операционных системах семейства NT.

После заголовка должен идти начальный блок, с которого будет устанавливаться данный INF-файл. Стало традицией, что этот блок должен иметь название `DefaultInstall`. К тому же блок именно с таким заголовком ищет система при установке INF-файла с помощью команды Установить его контекстного меню. Если же предполагается, что создаваемый INF-файл не должен вызываться с помощью контекстного меню (а только с использованием команды `rundll32.exe setupapi.dll, InstallHINFSection`), то начальный блок можно указать любой.

В начальном блоке могут содержаться различные ключевые слова, указывающие на другие блоки INF-файла, с помощью которых выполняется работа с реестром и файловой системой Windows XP.

Создание ветвей реестра

Например, в начальном блоке может находиться ключевое слово `AddReg`, указывающее на блоки INF-файла, описывающие добавляемые или измененяемые параметры и ветви реестра. Рассмотрим формат этого ключевого слова на примере листинга 15.1. В данном листинге приведен пример редактирования `DWORD`-параметра `AutoRun` из ветви реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Cdrom`. В примере этому параметру присваивается значение `0`, то есть отключается возможность автоматического запуска дисков.

Листинг 15.1. [illegible]

```
[version]
Signature="$CHICAGO$"

[DefaultInstall]
AddReg=AR_off

[AR_off]
HKLM, "SYSTEM\CurrentControlSet\Services\Cdrom","Autorun",0x10001,0
```

Как видно из листинга 15.1, в блоке для редактирования ветвей реестра (в ключевом слове `AddReg` можно через запятую указать несколько блоков для редактирования ветвей реестра) описываются сами ветви реестра и параметры, в них изменяемые. Формат их описания таков:

«[illegible]»,«[illegible]»,«[illegible]»,«[illegible]»,«[illegible]»

Рассмотрим этот формат подробнее.

- Êîðíåâîé ðàçäåë — здесь содержится ключевое слово, определяющее корневой раздел реестра, в котором расположен изменяемый параметр. Возможны следующие значения:
  - HKCU — определяет корневой раздел HKEY_CURRENT_USER;
  - HKLM — HKEY_LOCAL_MACHINE;
  - HKU — HKEY_USERS;
  - HKCR — HKEY_CLASSES_ROOT;
  - HKCC — HKEY_CURRENT_CONFIG.
- Âåòâü ðååñòðà — определяет остальной путь к ветви реестра, включающей в себя изменяемый параметр. Если ветвь реестра содержит пробелы, то ее нужно взять в кавычки.
- Ïàðàìåòð — указывает изменяемый параметр реестра. Если название параметра содержит пробелы, то его нужно взять в кавычки. Если название параметра указано не будет, то будет изменено значение параметра (Ïî óìîë÷àíèþ).
- Ôëàã — определяет как тип параметра, так и в некоторых случаях дополнительные сведения о том, что же нужно делать с аналогичным параметром в реестре, если он уже существует. Флаг представляет собой битовую маску. Рассмотрим некоторые состояния этой битовой маски.
  - 0õ00000000 — тип REG_SZ.
  - 0õ00000001 — REG_BINARY.
  - 0õ00010000 — REG_MULTI_SZ.
  - 0õ00020000 — REG_EXPAND_SZ.
  - 0õ00010001 — DWORD.
  - 0õ00020001 — NONE.
  - 0õ00000002 — если изменяемый параметр уже существует в реестре, то изменять его значение запрещено.
  - 0õ00000004 — удалить раздел или параметр из реестра. Иными словами, в INF-файле можно обойтись даже без специального ключевого слова для описания блока удаления, который будет рассмотрен далее. Удалить параметр можно и с помощью блока редактирования параметров.
  - 0õ00000008 — только для параметров REG_MULTI_SZ-типа. Указанное в строке редактирования параметра значение не заменяет существующее значение, а добавляется к существующему значению параметра.
  - 0õ00000010 — создать раздел, но игнорировать создание или редактирование указанного в строке параметра. Вообще, если посмотреть на возможные значения данного флага, то можно подумать, что Microsoft намеревается создать целый язык сценария с условными значениями и переменными

для INF-файла. Иначе зачем вообще нужны два только что рассмотренных значения флага, если аналогичные действия можно выполнить и без их использования?

- 0õ00000020 — изменить значение параметра, только если данный параметр уже существует в реестре.

■ Çíà÷åíèå ïàðàìåòðà — определяет новое значение параметра.

Теперь рассмотрим пример INF-файла, добавляющего в реестр значения параметров. Пример, отображенный в листинге 15.2, является частью стандартного INF-файла Windows XP, предназначенного для настройки отключения автозапуска дисков для разных типов приводов компакт-дисков. В примере параметру Autorun присваивается значение только в том случае, если он не существует в реестре. А значение параметра AutoRunAlwaysDisable, имеющего тип REG_MULTI_SZ, формируется в несколько приемов, чтобы обеспечить хранение значений параметра в разных строках.

Листинг 15.2.

```
[version]
Signature="$CHICAGO$"

[DefaultInstall]
AddReg=autorun_addreg

[autorun_addreg]
HKLM,"System\CurrentControlSet\Services\cdrom","AutoRun",0x00010003,1
HKLM,"System\CurrentControlSet\Services\cdrom","AutoRunAlwaysDisable", 0x00010008,
"NEC     MBR-7   "
HKLM,"System\CurrentControlSet\Services\cdrom","AutoRunAlwaysDisable", 0x00010008,
"NEC     MBR-7.4 "
HKLM,"System\CurrentControlSet\Services\cdrom","AutoRunAlwaysDisable", 0x00010008,
"PIONEER  CHANGR  DRM-1804X"
HKLM,"System\CurrentControlSet\Services\cdrom","AutoRunAlwaysDisable", 0x00010008,
"PIONEER  CD-ROM  DRM-6324X"
HKLM,"System\CurrentControlSet\Services\cdrom","AutoRunAlwaysDisable", 0x00010008,
"PIONEER  CD-ROM  DRM-624X  "
HKLM,"System\CurrentControlSet\Services\cdrom","AutoRunAlwaysDisable", 0x00010008,
"TORiSAN  CD-ROM  CDR_C36"
```

Удаление ветвей реестра

Параметр или ветвь реестра можно не только добавить в реестр, но и удалить из него. Для этого применяется ключевое слово DelReg, указывающее на блок INF-файла, содержащий сведения о ветвях реестра и параметрах, которые нужно удалить. Несмотря на то, что флаг для редактирования параметров позволяет также и удалять параметры, для их удаления рекомендуется все-таки использовать клю-

чевое слово, так как это более наглядно и позволяет легче понять принцип работы INF-файла.

Рассмотрим пример удаления параметра. В этом примере из реестра удаляется ветвь `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\lanmanserver\ Shares`, чтобы отключить все пользовательские общедоступные папки. При этом сначала нужно удалить все расположенные в данной ветви разделы, чтобы можно было удалить саму ветвь реестра. Как правило, ветвь для хранения сведений об общедоступных папках содержит только один раздел — `Security`.

ПРИМЕЧАНИЕ

Вообще-то можно удалять и ветви реестра, хранящие другие разделы, но лучше перестраховаться. На памяти у автора были случаи, когда ветвь реестра не удалялась из-за того, что в ней содержались вложенные разделы.

15.3. [illegible]

```
[version]
Signature="$CHICAGO$"

[DefaultInstall]
DelReg=shared_del

[shared_del]

HKLM,"SYSTEM\CURRENTCONTROLSET\SERVICES\LANMANSERVER\SHARES\Security"
HKLM,"SYSTEM\CURRENTCONTROLSET\SERVICES\LANMANSERVER\SHARES"
```

Как можно заметить, содержимое блока для удаления ветвей и параметров реестра похоже на содержимое блока редактирования ветвей и параметров реестра. Строка для удаления ветви или параметра имеет следующий формат:

«[illegible]»,«[illegible]»,«[illegible]»,«[illegible]»,«[illegible]»

- `Êîðíåâîé ðàçäåë` — указывает на корневой раздел, в котором расположен удаляемый параметр или ветвь реестра.
- `Âåòâü ðååñòðà` — определяет удаляемую ветвь реестра или ветвь, в которой хранится удаляемый параметр.
- `Ïàðàìåòð` — определяет название удаляемого параметра. Если параметр отсутствует, то предполагается, что удаляться будет конечный раздел указанной ветви реестра.
- `Ôëàã îïåðàöèè óäàëåíèÿ` — может принимать следующие значения:
  - `0õ00002000` — удалить весь конечный раздел указанной ветви;
  - `0õ00004000` — произвести указанные изменения в 32-разрядном реестре;

- 0õ00018002 — удаляет из параметра все строки, соответствующие примеру для удаления.

- Ïðèìåð äëÿ óäàëåíèÿ — определяет строку значения параметра, имеющего REG_MULTI_SZ-тип, все соответствия которой должны быть удалены из параметра.

Редактирование отдельных битов значения параметра

Это довольно интересная и, можно сказать, уникальная возможность, с помощью которой можно изменить отдельный бит параметра, не изменяя другие его биты. Для реализации этой возможности применяется ключевое слово BitReg, указывающее на блок INF-файла, содержащий сведения об изменяемых битах параметров. При этом блок INF-файла должен включать в себя строки следующего формата:

«□□□□□□□□□□□□□□»,«□□□□□□□□□□□□□□»,«□□□□□□□□»,«□□□□□□□□□□□□□□»,«□□□□□□□□□□□□□□»,
«□□□□□□□□□□□□□□□□□□□□»

- Êîðíåâîé ðàçäåë, âåòâü ðååñòðà и ïàðàìåòð были рассмотрены ранее.

- Ôëàã îïåðàöèè — может принимать следующие значения:
  - 0õ00000000 — сбросить указанный бит;
  - 0õ00000001 — установить указанный бит;
  - 0õ00040000 — выполнить эти операции в 32-разрядном реестре.

- Ìàñêà îïåðàöèè — определяет биты в значении параметра, которые должны быть модифицированы. Другими словами, маска должна состоять из восьми нулей или единиц (определяют 8 бит одного байта значения параметра). Все биты, на месте которых в маске указана единица, будут модифицироваться в зависимости от флага операции. Маска указывается в виде битовой маски.

- Íîìåð áàéòà çíà÷åíèÿ ïàðàìåòð — указывает на байт значения параметра, к которому будет применяться маска и биты которого будут модифицироваться. При этом номер байта зависит от типа параметра. Для параметров DWORD-типа самый старший байт имеет номер 0, а для параметров REG_BINARY-типа номер 0 имеет самый младший байт.

Для примера попробуем изменить отдельные биты параметра Attributes контекстного меню Корзины. После данной модификации в контекстном меню Корзины будут команды Переименовать, Свойства и Удалить. Такие команды, как Копировать, Вырезать, Вставить, будут удалены из контекстного меню Корзины (если они там присутствуют). В результате применения приведенного INF-файла значение DWORD-параметра Attributes станет равным 0õ????0070.

ПРИМЕЧАНИЕ

Заметьте, что сначала желательно сбрасывать биты отдельного байта, а потом уже устанавливать другие биты этого байта.

Если в ветви реестра HKEY_CLASSES_ROOT\CLSID\{645FF040-5081-101B-9F08-00AA002F954E}\ShellFolder присутствует параметр, имеющий DWORD-тип, CallForAttributes, то ему будет присвоено значение 0 (если в ветви данного параметра не существует, то он и не будет создан).

Листинг 15.4.

```
[version]
Signature="$CHICAGO$"

[DefaultInstall]
BitReg=RecicleBit
AddReg=CallAttrOff

[CallAttrOff]
HKCR,"CLSID\{645FF040-5081-101B-9F08-00AA002F954E}\
ShellFolder",CallForAttributes,0x00010021,0

[RecicleBit]
HKCR,"CLSID\{645FF040-5081-101B-9F08-00AA002F954E}\
ShellFolder",Attributes,0,0xff,0
HKCR,"CLSID\{645FF040-5081-101B-9F08-00AA002F954E}\
ShellFolder",Attributes,0,0xff,1
HKCR,"CLSID\{645FF040-5081-101B-9F08-00AA002F954E}\
ShellFolder",Attributes,1,0x70,0
```

Создание службы

Еще одной оригинальной возможностью, которой обладают INF-файлы, является упрощенное создание служб на компьютере. Для этого применяется не только ключевое слово AddService, но и специальный стандартный блок INF-файла [DefaultInstall.Services] (то есть к блоку по умолчанию добавляется строка .Services). При этом следует сказать, что этот блок не заменяет стандартный, а дополняет его. Иными словами, если в INF-файле будет два блока, то сначала выполнятся ключевые слова блока [DefaultInstall], а потом блока [DefaultInstall.Services].

Так как вам уже известно, как хранятся сведения о службе в реестре, то будет не сложно понять пример создания службы. По этой причине сначала посмотрим на листинг 15.5, который содержит часть INF-файла, регистрирующего в системе службу Восстановление системы.

Листинг 15.5.

```
[version]
Signature="$CHICAGO$"

[DefaultInstall.Services]
AddService=sr,,SRFlt_service,SRFlt_event
```

```
[SRFlt_service]
DisplayName      ="□□□□□□□□"
ServiceType      = 2
StartType        = 0
ErrorControl     = 1
ServiceBinary    = %12%\sr.sys
LoadOrderGroup   = "FSFilter System Recovery"

[SRFlt_event]
AddReg=SRFlt_event_addreg
DelReg=SRFlt_event_delreg
```

Ключевое слово `AddService`, в отличие от большинства других, содержит не только название блока INF-файла, описывающего службу, но и некоторые другие сведения. Формат этого ключевого слова следующий:

«□□□□□□□□□□□□□□»,«□□□□□□□□», «□□□□INF-□□□□□»

- `Íàçâàíèå ñëóæáû` — определяет название раздела в ветви системного реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services`, в который будет заноситься информация о службе. Как известно, название этого раздела является и названием службы.
- `Òèï ñëóæáû` — указывает, является ли данная служба самостоятельной. Может принимать значения `0x01`, `0x2` и `0x3`.
- `Áëîê INF-îáéëà` — определяет один или несколько блоков INF-файла (в этом случае они пишутся через запятую), в которых определены сведения о службе. В листинге 15.5 первый блок содержит информацию о службе, а второй блок регистрирует возможность записи в системные журналы Windows (оснастка Просмотр событий).

Теперь рассмотрим блок INF-файла для регистрации службы. Он может включать в себя следующие ключевые слова.

- `DisplayName` — имя службы, отображаемое в оснастке `services.msc`.
- `Description` — описание службы, отображаемое в оснастке `services.msc`.
- `ServiceType` — тип службы. Значение этого параметра соответствует уже рассмотренному значению параметра реестра `Type` (см. часть 2).
- `StartType` — режим запуска службы. Значение этого параметра соответствует уже рассмотренному значению параметра реестра `Start`.
- `ErrorControl` — действие при возникновении ошибки при запуске службы. Значение этого параметра соответствует уже рассмотренному значению параметра реестра `ErrorControl`.
- `ServiceBinary` — путь и имя файла службы. Значение данного ключевого слова заносится в уже рассмотренный параметр `ImagePath` реестра.

- `Dependencies` — определяет службы, необходимые для работы данной службы. Эти значения заносятся в параметр реестра `DependOnService`.
- `LoadOrderGroup` — определяет группу, к которой принадлежит данная служба. Эти значения заносятся в параметр реестра `Group`.

Удаление службы

Для удаления службы предназначено ключевое слово `DelService`, которое также указывается в блоке, названном в формате `[«áëîê óäàëåíèÿ».Services]`. Удаление службы выполняется намного проще — просто указывается имя службы, которую нужно удалить (название раздела реестра, в котором служба описана).

Листинг 15.6.

```
[version]
Signature="$CHICAGO$"

[Uninstall.Services]
DelService=sr
```

В листинге 15.6 ключевое слово `DelService` указано в блоке `[Uninstall.Services]`, так как именно блок `[Uninstall]` используется в примере при вызове INF-файла для удаления службы с помощью команды `rundll32.exe`.

Копирование файлов

Копирование в INF-файлах реализовано наиболее сложно. Для его выполнения необходимо не только использовать ключевое слово `CopyFiles`, но и, кроме того, создать три отдельных блока INF-файла: `DestinationDirs`, `SourceDisksFiles` и `SourceDisksNames`. Рассмотрим простой пример.

Листинг 15.7.

```
[version]
Signature = $CHICAGO$

[DefaultInstall]
CopyFiles = INFcopy

[INFcopy]
1.INF

[DestinationDirs]
INFcopy = 17

[SourceDisksFiles]
1.INF = 55
```

```
[SourceDisksNames]
55 = "[illegible]", , 1
```

В этом примере для описания копируемых файлов используется блок INF-файла `[INFcopy]`. Ссылка на него содержится в ключевом слове `CopyFiles`. В этом блоке находятся только названия файлов, которые должны быть скопированы (если файлов несколько, каждый из них пишется в отдельной строке). Кроме блока `[INFcopy]`, файл обязательно должен хранить и три других стандартных блока INF-файлов.

Блок `[DestinationDirs]` определяет каталоги, в которые должны копироваться файлы, описанные в отдельных блоках INF-файла, предназначенных для описания копируемых файлов. При этом все файлы, описанные в одном блоке INF-файла, должны копироваться только в один каталог (то есть если бы в нашем блоке `[INFcopy]` описывалось два файла, например `1.INF` и `2.INF`, то они оба должны были бы скопироваться в один каталог). Формат определения каталога, в который будут копироваться файлы, следующий:

«[illegible]»=«[illegible]»,«[illegible]»

Числовой номер каталога представляет собой число, идентифицирующее одну из системных папок. Рассмотрим некоторые из этих чисел, а также каталоги, которые они определяют.

- `10` — `%systemroot%`.
- `11` — `%systemroot%\system32`.
- `17` — `%systemroot%\Inf`.
- `53` — `%userprofile%`.
- `54` — `%systemdrive%`.
- `1` — говорит о том, что необходимо использовать абсолютный путь, который должен быть записан после числового номера (через запятую). Абсолютным называется путь, который не начинается с числового номера каталога. Например, числовым является путь `d:\test\1`.

С помощью блока `[SourceDisksFiles]` осуществляется присваивание каждому копируемому файлу уникального идентификатора, он будет определять в блоке `[SourceDisksNames]` название диска, с которого копируется файл. Формат строк этого блока следующий:

«[illegible]»=«[illegible]»

Блок `[SourceDisksNames]` определяет диски, на которых находятся копируемые файлы. Формат строк этого блока таков:

«[illegible]»=«[illegible]»,«[illegible]»,«[illegible]»

Метка диска отображается в диалоге копирования файлов. Эта метка также отображается в диалоге, выводимом, если система не может найти указанные для копирования файлы. В этом случае система попросит вас вставить диск, название которого как раз и указано в блоке `[SourceDisksNames]`.

Удаление файлов

Удаление файлов напоминает их копирование, поэтому рассмотрим этот процесс лишь поверхностно — в подсказках к листингу 15.8. Для определения блоков, содержащих описания удаляемых файлов, используется ключевое слово `DelFiles`.

Листинг 15.8.

```
[version]
Signature = $CHICAGO$

;
rundll32
[DefaultUninstall]
DelFiles = INFdel

;
vfolder.INF
[INFdel]
vfolder.INF

;
;
[DestinationDirs]
INFdel = 17

[SourceDisksFiles]
vfolder.INF = 55

[SourceDisksNames]
55 = "",,1
```

Создание ярлыка файл в меню Пуск

Это еще одна возможность, которой можно воспользоваться благодаря INF-файлу. С помощью ключевого слова `ProfileItems` можно определить блок INF-файла, в котором описывается создаваемый в меню Пуск ярлык. Рассмотрим пример, с помощью которого создается ярлык программы Восстановление системы в меню Пуск (Стандартные ▸ Служебные).

Листинг 15.9.

```
[version]
Signature="$CHICAGO$"
```

```
[DefaultInstall]
ProfileItems=SRProfile

[SRProfile]
Name="□□□□□□□□□□□□□□□□□□□□□"
CmdLine=11,restore,rstrui.exe
SubDir="□□□□□□□□□□□\□□□□□□□□□"
InfoTip="□□□□□□□□□□□□,□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□"
DisplayResource="%systemroot%\system32\restore\rstrui.exe",2048
ProfileItems=SRProfile
```

Блок для описания ярлыка в меню Пуск может содержать следующие ключевые слова.

- `Name` — определяет название создаваемого в меню Пуск ярлыка.
- `CmdLine` — указывает путь к программе, для которой создается ярлык. Формат данного ключевого слова немного необычен: «`÷èñëîâîé ïóòü ê ñòàíäàðò-íîìó êàòàëîãó, ðàçìåùåíèå ôàéëà, íàçâàíèå ôàéëà`».
- `SubDir` — определяет расположение ярлыка в меню Пуск. Если точнее, то данное ключевое слово определяет дальнейший путь к каталогу, который будет содержать ярлык на программу. Дальнейший путь добавляется к пути `%systemdrive%\Documents and Settings\All Users\Ãëàâíîå ìå-íþ\Ïðîãðàììû`.
- `WorkingDir` — указывает рабочие каталоги для программы, на которую создается ярлык (как правило, это ключевое слово не указывается).
- `InfoTip` — определяет подсказку, отображаемую при наведении указателя на созданный в меню Пуск ярлык.

Расширенные INF-файлы

Расширенные INF-файлы отличаются от обычных одной строкой, которую нужно добавить в блок `[version]`. Ниже приведен пример блока версии для расширенного INF-файла.

```
[Version]
Signature="$CHICAGO$"
AdvancedINF=2.5,"□□□□□□□□□□□□□□□□□□□□□□□□,□□□□□□□□□□□□□□□□□□□□□□□□□
□□□□□□□□□□ advpack.dll"
```

Для работы с расширенными INF-файлами необходима библиотека `advpack.dll`, по умолчанию расположенная в каталоге `%systemroot%`. Если перед началом выполнения расширенного INF-файла система не найдет данной библиотеки, то она выведет сообщение, указанное в ключевом слове `AdvancedINF` после запятой. К тому же, как известно, расширенные INF-файлы вызываются с помощью команды `rundll32.exe ADVPACK.dll, LaunchINFSectionEx «èìÿ ôàéëà»,`

«èìÿ ðàçäåëà», «èìÿ cab-ôàéëà», «ôëàã». Тогда как обычные INF-файлы вызываются с помощью следующей командой: `rundll32.exe setupapi.dll, InstallHINFSection «ðàçäåë â ôàéëå äëÿ íà÷àëà óñòàíîâêè», «ôëàã», «ïóòü è èìÿ ôàéëà».`

Запуск программ до и после установки

Расширенные INF-файлы поддерживают такую возможность, как запуск команд перед установкой INF-файла или после нее. При этом для указания блока, описывающего программы, запускаемые перед установкой INF-файла, используется ключевое слово `RunPreSetupCommands`. Для указания блока, описывающего программы, запускаемые после установки INF-файла, используется ключевое слово `RunPostSetupCommands`. Посмотрим на пример использования этих ключевых слов.

Листинг 15.10. [illegible]INF

```
[version]
Signature = $CHICAGO$
AdvancedINF=2.5,"[illegible],[illegible]
[illegible]advpack.dll"

[DefaultInstall]
RunPreSetupCommands=RunPre
RunPostSetupCommands=RunPost

[RunPre]
calc.exe
cmd.exe

[RunPost]
"rundll32.exe IEAKENG.dll, DoReboot"
```

Приведенный в листинге 15.10 INF-файл перед своей установкой вызывает Калькулятор, а потом — стандартный командный интерпретатор Windows. После своей установки он вызовет диалоговое окно с вопросом о перезагрузке компьютера. Конечно, это окно принадлежит Internet Explorer, но зато оно работает.

ПРИМЕЧАНИЕ

Вопрос о перезагрузке компьютера можно было вызвать и с помощью флагов вызова INF-файла, но в листинге специально были использованы команды rundll32.exe.

Не забывайте также, что данный INF-файл не выполнится после выбора команды Установить контекстного меню INF-файла. Для его установки придется воспользоваться командой `RunDll32 advpack.dll, LaunchINFSection d:\1.INF, DefaultInstall`. Здесь `d:\1.INF` соответствует пути к INF-файлу и его названию.

Вывод сообщения перед установкой

Перед установкой INF-файла или после нее можно вывести окно сообщения с произвольным текстом. Диалог сообщения, выводимый перед установкой, позволяет эту установку отменить. Он содержит две кнопки — ОК и Отмена. Если нажать кнопку Отмена, то установка INF-файла будет отменена. Диалог сообщения, выводимый после установки, является информационным и имеет только одну кнопку — ОК.

Чтобы вывести диалоговое окно перед установкой, необходимо воспользоваться ключевым словом `BeginPrompt` (окно сообщения будет выведено до вызова программ, описанных ключевым словом `RunPreSetupCommands`), а чтобы вывести окно сообщения после установки, нужно воспользоваться ключевым словом `BeginPrompt`. В листинге 15.11 приведен простой пример использования как окна сообщения, выводимого перед установкой, так и окна сообщения, выводимого после установки.

Листинг 15.11.

```
[version]
Signature = $CHICAGO$
AdvancedINF=2.5,"
advpack.dll"

[DefaultInstall]
RunPreSetupCommands = RunPre
BeginPrompt=BeginText
EndPrompt=EndText

[RunPre]
cmd.exe

[BeginText]
Prompt="
"
Title="
"

[EndText]
Prompt="
"
```

Блок для описания сообщения, выводимого перед установкой INF-файла (в данном случае `BeginText`) содержит следующие ключевые слова.

- `Prompt` — определяет саму строку выводимого сообщения.
- `Title` — указывает заголовок окна сообщения. Следует учитывать, что данный заголовок будет использоваться не только для окна сообщения, отображаемого перед установкой, но и для окна сообщения, отображаемого после установки.

Дополнительные возможности

Выше были рассмотрены лишь основы работы с INF-файлами. Кроме приведенных ключевых слов, INF-файлы поддерживают многие другие, но если начать их описывать, то понадобится отдельная книга. Закончим на этом рассказ о ключевых словах INF-файлов. Теперь рассмотрим несколько примеров не совсем стандартного использования INF-файлов. Без описания этих примеров рассказ о возможностях INF-файлов был бы не полон.

Работа с диалогом Установка и удаление программ

Одной из интересных возможностей INF-файлов является возможность их использования как для добавления команды в диалог Установка и удаление программ, так и для выполнения процесса деинсталляции при помощи диалога Установка и удаление программ. Рассмотрим простой пример использования INF-файлов для создания так называемого зацикленного элемента диалога Установка и удаление программ, который применяется для включения и отключения возможности автоматического запуска дисков. Принцип работы приведенного ниже сценария прост. При первом своем вызове он копирует себя в папку `%systemroot%\INF`, отключает автоматический запуск дисков, а также добавляет возможность включения автоматического запуска дисков в диалог Установка и удаление программ. После удаления данного INF-файла с помощью диалога Установка и удаление программ происходит включение автоматического запуска дисков, а также создание в диалоге Установка и удаление программ новой строки, с помощью которой можно опять отключить автоматический запуск диска. Другими словами, создается постоянный цикл. Конечно, пример с изменением значения одного параметра довольно спорен, ведь намного проще было бы добавить свой флажок в один из стандартных диалогов Windows, поддерживающих добавление в свои списки новых элементов. Но если необходимо при установке параметра также выполнять какие-либо команды или модифицировать сразу несколько параметров (например, создать несколько разновидностей настройки оболочки Windows, а потом переходить между ними), то данный способ использования INF-файлов может быть незаменим (рис. 15.1).

15.12.

```
[version]
Signature = $CHICAGO$

[DefaultInstall]
AddReg = AutoRunOff, InstallInf
CopyFiles = INFcopy

;                                %PATH%
;            -
```

```
;
;
;
;
;
;
;
[AutoRunOff]
HKLM, %PATH%, AutoRun, 0x00010001, 0

;
;
[InstallInf]
HKLM, %DISPLAY% autorunOnOff,,,
HKLM, %DISPLAY% autorunOnOff, DisplayName,,%NAME_ON%
HKLM, %DISPLAY% autorunOnOff, UninstallString,,\
"rundll32 setupapi, InstallHINFSection DefaultUninstall 132 %17%\primer.INF"

[DefaultUninstall]
AddReg = AutoRunOn

[AutoRunOn]
HKLM, %PATH%, AutoRun, 0x00010001, 1
HKLM, %DISPLAY% autorunOnOff, DisplayName,,%NAME_OFF%
HKLM, %DISPLAY% autorunOnOff, UninstallString,,\
"rundll32 setupapi, InstallHINFSection DefaultInstall 132 %17%\primer.INF"

[INFcopy]
primer.INF

[DestinationDirs]
INFcopy = 17

[SourceDisksFiles]
primer.INF = 55

[SourceDisksNames]
55 = %DISKNAME%, , 1

;
[Strings]
PATH = "SYSTEM\CurrentControlSet\Services\Cdrom"
DISPLAY = "SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall\"
```

NAME_ON="[illegible]"
NAME_OFF="[illegible]"
DISKNAME = "parad0x-des1Gn"

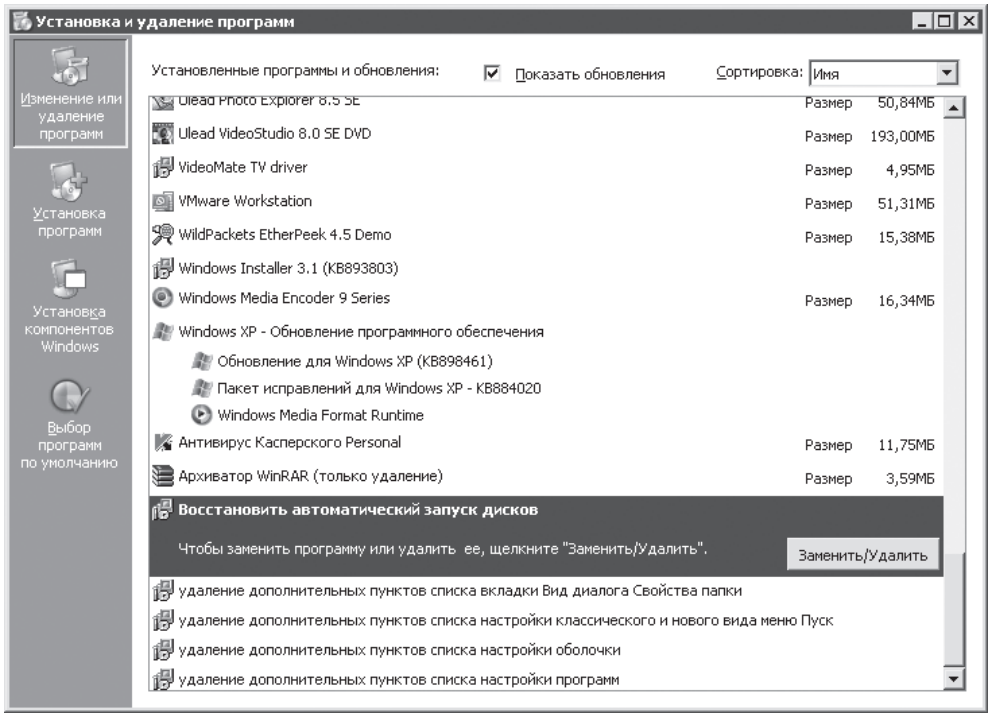


Рис. 15.1. Создание команды в диалоге Установка и удаление программ

Диалог установки компонентов Windows

Отдельной частью окна Установка и удаление программ является диалог Установка компонентов Windows, который вызывается нажатием соответствующей кнопки. Как правило, данный диалог включает в себя сведения не обо всех компонентах Windows, доступных на компьютере. Например, с его помощью нельзя удалить с компьютера стандартные игры, такие как Сапер или Паук. Это происходит потому, что по умолчанию возможность удаления или установки игр скрыта из диалога установки компонентов Windows. Список содержимого диалога Установка компонентов Windows описывается в файле `sysoc.INF`, расположенном в каталоге `%systemroot%\INF`. Данный файл включает в себя блок INF-файла `[Components]`. Блок содержит наборы строк, определяющих, будет ли отображаться в диалоге отдельный компонент операционной системы Windows XP. Строки имеют следующий формат:

«[illegible]»=«[illegible]»,«[illegible]»,
«NF[illegible]»,«[illegible]»,«[illegible]»,«[illegible]»

Следующие элементы являются наиболее важными.

- `INF-ôàéë, îïèñûâàþùèé êîìïîíåíòû` — описывает разделы, отображаемые в диалоге Установка и удаление программ. Именно такой INF-файл сейчас и будет создан.
- `Îòîáðàæåíèå êîìïîíåíòîâ` — если в данном поле будет стоять слово `HIDE`, то данный компонент не будет отображаться в диалоге дополнительных компонентов Windows. Чтобы компонент отображался в диалоге, необходимо чтобы это поле было пустым.
- Например, чтобы добавить в диалог компонентов возможность удаления игр, необходимо в строке, начинающейся с идентификатора `games`, удалить строку `HIDE`.

На рис. 15.2 показан пример содержимого этого файла.

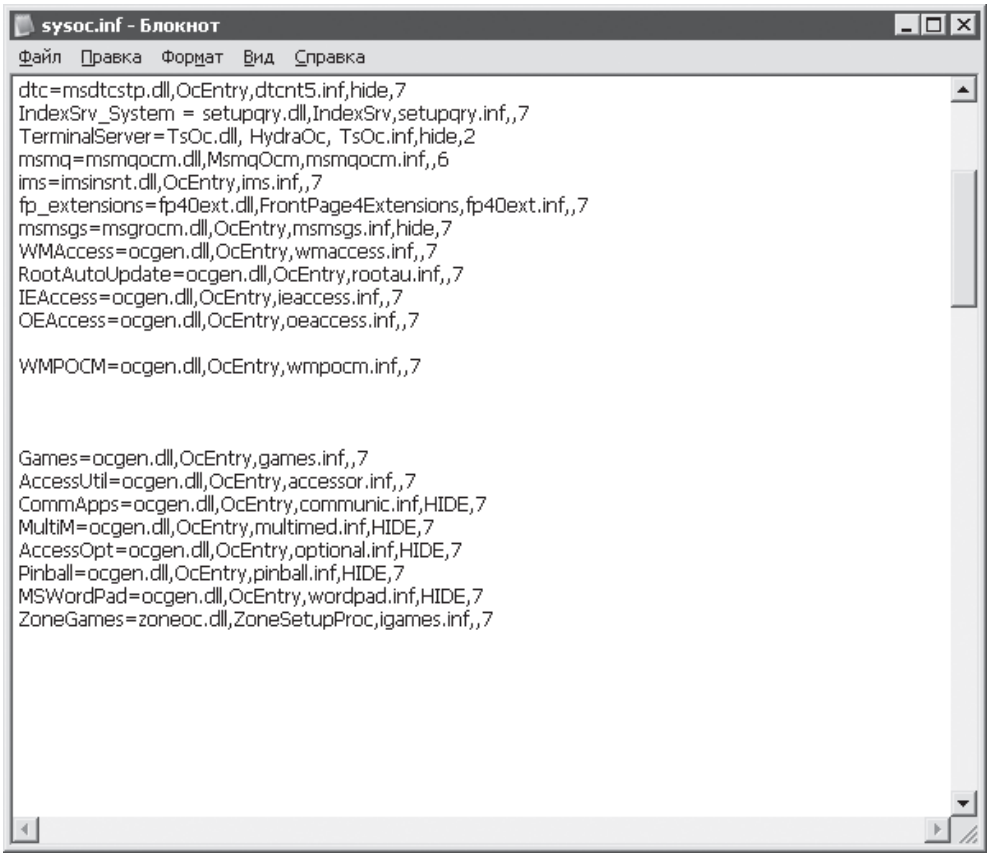


Рис. 15.2. Содержимое файла sysoc.INF

Другой возможностью является добавление в данный диалог своих компонентов. Для этого необходимо сначала создать INF-файл, описывающий новые компонен-

ты, а потом создать в файле `sysoc.inf` ссылку на созданный INF-файл. Сначала посмотрим на простой пример INF-файла. В этом примере сразу создается корневой раздел, отображаемый непосредственно в диалоге установки компонентов Windows, два вложенных в этот раздел подраздела, а также еще два раздела, которые и будут устанавливать или удалять компоненты. В нашем случае эти разделы будут просто скрывать (или отображать, в зависимости от состояния флажка) версию Windows на Рабочем столе, а также скрывать (или отображать) название значка Мой компьютер.

Пример 15.13. [illegible]

```
[version]
signature="$Chicago$"

[Optional Components]
TopMain
Interface
Settings
HTMcomputer
Versus

[TopMain]
OptionDesc      =[illegible]
Tip             =[illegible]
IconIndex       =0

[Interface]
OptionDesc      =[illegible]Windows
Tip             =[illegible]
IconIndex       =4
Parent          = TopMain

[Settings]
OptionDesc      =[illegible]Windows
Tip             =[illegible]
IconIndex       = 16
Parent          = TopMain

[HTMcomputer]
OptionDesc      =[illegible]
Tip             =[illegible]
```

```
IconIndex          = 0
Parent             = Interface
AddReg             = AddHTM
Uninstal           = HTMUn
Modes              = 0,1,2,3

[Versus]
OptionDesc         =[illegible]
Tip                =[illegible]
[illegible]
IconIndex          =4
Parent             = Interface
AddReg             = AddVersion
Uninstall          = VersUn
Modes              = 0,1,2,3

[AddHTM]
HKCR, "CLSID\{20D04FE0-3AEA-1069-A2D8-08002B30309D}\ShellFolder",
WantsFORDISPLAY, 0x00010001, 1

[AddVersion]
HKCU, "Control Panel\Desktop", PaintDesktopVersion, 0x00010001, 1

[HTMUn]
DelReg = HTMDel

[VersUn]
AddReg = VersDel

[HTMDel]
HKCR, "CLSID\{20D04FE0-3AEA-1069-A2D8-08002B30309D}\ShellFolder",
WantsFORDISPLAY

[VersDel]
HKCU, "Control Panel\Desktop", PaintDesktopVersion, 0x00010001, 0
```

Формат INF-файла для создания компонентов в диалоге установки Windows немного отличается от обычного формата INF-файлов. Во-первых, основной блок INF-файла обязательно должен называться `[Optional Components]`, иначе

INF-файл работать не будет. Блок включает в себя список других блоков данного INF-файла. Каждый из описанных в `[Optional Components]` блоков определяет один раздел в диалоге установки компонентов Windows.

Блок описания раздела установки компонента содержит следующие ключевые слова.

- `OptionDesc` — определяет название раздела, отображаемого в диалоге Установка компонентов Windows.
- `Tip` — указывает подсказку для раздела, отображаемого в нижней части диалога Установка компонентов Windows.
- `IconIndex` — определяет индекс значка, отображаемого напротив данного раздела в диалоге Установка компонентов Windows. Например, идентификатор `0` определяет значок компьютера, идентификатор `2` — монитора, `14` — принтера.
- `Parent` — указывает название блока INF-файла, описывающего раздел, который будет родителем для нашего раздела в диалоге установки компонентов Windows. Если данное ключевое слово отсутствует, то наш раздел будет отображаться непосредственно в диалоге установки конфигурации Windows.
- `Uninstall` — определяет блок INF-файла, вызываемый для данного компонента при установке созданного раздела (устанавливать можно только разделы, входящие непосредственно в список диалога установки компонентов (корневые), а не отдельно разделы, вложенные в корневой), если флажок напротив данного компонента будет снят.

В блоке установки компонента можно также пользоваться такими стандартными ключевыми словами, как `AddReg`, `BitReg`, `DelReg`, `CopyFiles` и т. д. Все блоки, описанные в этих ключевых словах, будут выполняться при установке флажка напротив соответствующего компонента.

Теперь добавим ссылку на наш компонент в файл `sysoc.inf`. Для того чтобы так сделать, необходимо в блок `[Components]` добавить строку следующего вида: `hello=ocgen.dll,OcEntry,prim2.inf,,7`. Здесь `test2` определяет идентификатор присоединяемого INF-файла, а `prim.inf` является названием самого присоединяемого файла. Стоит еще сказать, что созданный INF-файл должен находиться в каталоге `%systemroot%\inf`.

ПРИМЕЧАНИЕ

В строке также указывается название библиотеки и функция из этой библиотеки, которая будет устанавливать наши компоненты. Как правило, нет никакой разницы между различными функциями различных библиотек. Главное, чтобы они умели работать с диалогом установки компонентов. Поэтому была выбрана первая попавшаяся в файле sysoc.inf библиотека и функция для нее.

На рис. 15.3 показан результат добавления компонентов.

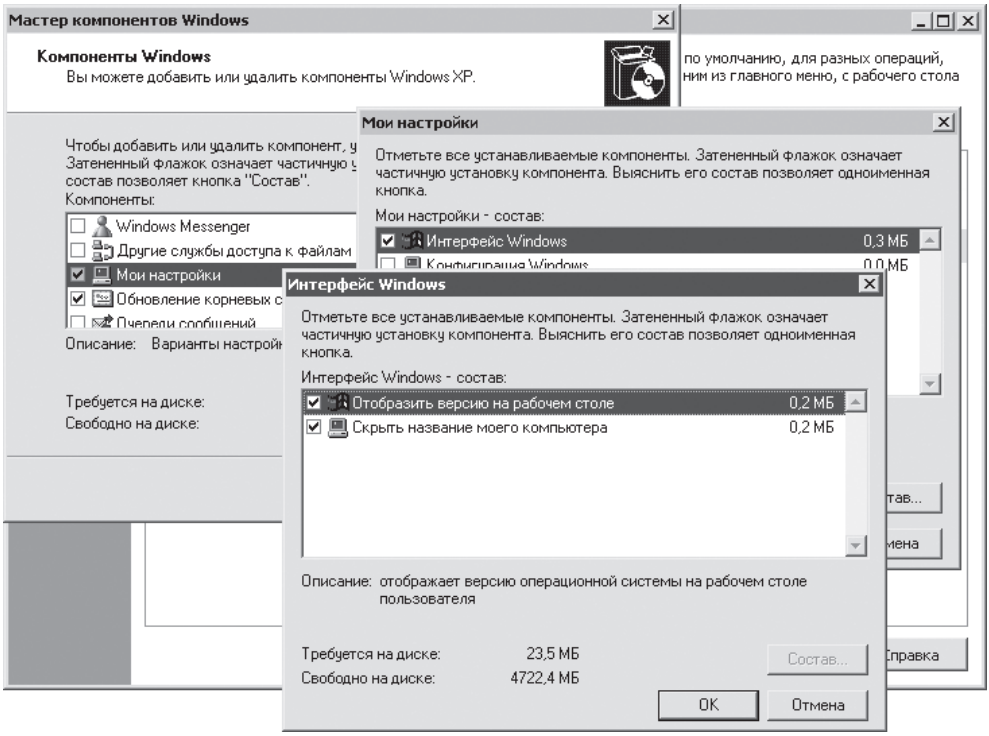


Рис. 15.3. Создание разделов диалога

INF-файл для логического диска

Еще одной возможностью, которую предоставляют INF-файлы, является добавление в контекстное меню логических дисков своих команд, а также изменение значка логического диска. Для этого необходимо создать специальный файл, который должен называться `autorun.inf`. Созданный INF-файл необходимо поместить в корень логического диска. Рассмотрим пример создания файла `autorun.inf`.

15.14. autoruninf

```
[autorun]
icon = E:\images\fotoo\Art_galery\p.BMP
shell = open
shell\RunPh\command = photoshop.exe
shell\RunPh = photoshop
```

Файл `autorun.inf` не имеет блока версии INF-файла, но должен иметь блок `[autorun]`. Содержимое этого блока довольно просто в понимании, если вы прочитали главу 8 о корневом разделе `HKEY_CLASSES_ROOT`. Как можно заметить,

ключевые слова, начинающиеся со слова `shell`, являются просто разделами реестра, которые должны быть добавлены к ветви `HKEY_CLASSES_ROOT\Drive` при отображении контекстного меню данного логического диска, а ключевое слово `icon` определяет значок диска. Тем не менее вспомним содержимое корневого раздела `HKEY_CLASSES_ROOT\Drive\shell` и опишем, что же конкретно делают ключевые слова данного файла.

- `shell = open` — добавляет в параметр `(Ïî óìîë÷àíèþ)` раздела `shell` строку `open`. Эта строка говорит о том, что по умолчанию при двойном щелчке на диске он должен открываться.
- `Shell\RunPh = Çàïóñòèòü photoshop` — добавляет в параметр `(Ïî óìîë÷àíèþ)` раздела `RunPh` строку `Çàïóñòèòü photoshop`. Эта строка определяет название команды в контекстном меню нашего диска.
- `Shell\RunPh\command = photoshop.exe` — добавляет в параметр `(Ïî óìîë÷àíèþ)` раздела `command` строку `photoshop.exe`. Эта строка определяет команду, которая будет выполняться при выборе из контекстного меню нашего диска соответствующей команды.

ПРИМЕЧАНИЕ

Для возможности работы файла autorun.inf необходимо, чтобы REG_BINARY-параметр NoDriveTypeAutoRun, расположенный в ветви реестра HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Policies\Explorer, был равен 0.

Как можно заметить, с помощью INF-файлов можно выполнить довольно много интересного. При этом рассмотренные возможности являются только каплей в море. Например, с помощью INF-файлов можно выполнить такие действия, как запуск или остановка служб, назначение прав на доступ к файлам, архивирование реестра и многое другое. Кроме того, INF-файлы постоянно усовершенствуются и еще неизвестно, что с их помощью можно будет выполнить завтра.

Глава 16

Сервер сценариев Windows

- **Реестр**
- **Файловая система**
- **Другие возможности**

При рассказе об инструментарии управления WMI уже рассматривались примеры создания сценариев сервера сценариев Windows с использованием WMI, сейчас же рассмотрим некоторые объекты, доступ к которым можно получить в Windows XP, а также методы для работы с ними. Глава будет оформлена как справочник, так как работа с объектами легче, чем работа с инструментарием WMI, тем более что в Интернете можно найти очень много примеров работы с объектами.

Реестр

Для доступа к реестру необходимо подключить объект `WshShell`. Для этого нужно воспользоваться следующим объявлением: `Set èìÿ_íîâîãî_îáúåêòà = WScript.CreateObject("WScript.Shell")`. Объект поддерживает следующие методы.

- `RegRead (ïóòü ê âåòâè ðååñòðà è èìÿ ïàðàìåòðà)` — считывает из реестра значение параметра, указанного в качестве входного значения. При этом если входное значение будет завершаться косой чертой (`\`), то будет считываться значение по умолчанию данной ветви реестра. Следует также напомнить, что путь к ветви реестра должен начинаться не с полного названия корневого раздела, а с его аббревиатуры. Например, возможны следующие аббревиатуры:
  - `HKCU` — соответствует корневому разделу `HKEY_CURRENT_USER`;
  - `HKLM` — `HKEY_LOCAL_MACHINE`;
  - `HKCR` — `HKEY_CLASSES_ROOT`.
- `RegWrite ïóòü ê âåòâè ðååñòðà è èìÿ ïàðàìåòðà, çíà÷åíèå ïàðàìåòðà, òèï ïàðàìåòðà` — редактирует значение существующего параметра реестра или создает новый параметр. При этом для его работы необходимо три входных значения, последнее из которых является аббревиатурой типа создаваемого параметра. Возможны следующие аббревиатуры:
  - `REG_SZ` — строковый параметр;
  - `REG_DWORD` — параметр `REG_DWORD`-типа;
  - `REG_BINARY` — параметр `REG_BINARY`-типа.

  Как и в методе `RegRead`, если путь к создаваемому параметру будет заканчиваться косой чертой, то будет изменяться значение параметра `(Ïî óìîë÷àíèþ)`.
- `RegDelete ïóòü ê âåòâè ðååñòðà è èìÿ ïàðàìåòðà, êîòîðûé ìû óäàëÿåì` — удаляет из реестра указанный во входящем значении параметр. При этом если входящее значение оканчивается косой чертой, то будет удалена вся ветвь.

Рассмотрим простой пример работы с данными методами. В нем будет создан параметр, а также отредактирован параметр `(Ïî óìîë÷àíèþ)`. Затем произойдет попытка считать созданные параметры, а после этого удаление сначала отдельного параметра, а потом всей ветви реестра, которая была создана ранее.

□□□□□□□□□□ 16.1. □□

```
set wshshell = WScript.CreateObject("WScript.Shell")

wshshell.RegWrite "HKCU\Primer_sozdania_vetvi\hello_world", "BIG_WORLD", "REG_SZ"
wshshell.RegWrite "HKCU\Primer_sozdania_vetvi\", "□□□□□□□□□ □□□□□□□□□ □□□□□□□□□□ □□
□□□□□□□□□□", "REG_SZ"

MsgBox wshshell.RegRead ("HKCU\Primer_sozdania_vetvi\hello_world")
MsgBox wshshell.RegRead ("HKCU\Primer_sozdania_vetvi\")

wshshell.RegDelete "HKCU\Primer_sozdania_vetvi\hello_world"
wshshell.RegDelete "HKCU\Primer_sozdania_vetvi\"
```

Файловая система

Для доступа к файловой системе используется следующий вызов: `Set èìÿ_îáúåêòà = CreateObject("Scripting.FileSystemObject")`. Объект поддерживает следующие методы.

- `BuildPath ("ïóòü ê êàòàëîãó", "íàçâàíèå ôàéëà")` — создает путь на основе указанного пути к каталогу и имени файла. Иными словами, он просто возвращает строку формата «`ïóòü ê êàòàëîãó» & "\" & «èìÿ ôàéëà»`.
- `CopyFile «ïóòü è èìÿ êîïèðóåìîãî ôàéëà», «ïóòü è èìÿ íîâîãî ôàéëà», «ôëàã ïåðåçàïèñè»` — копирует файл, указанный в первом входном параметре, туда, куда указывает второй входной параметр (и, если необходимо, меняет имя файла). При этом третий входной параметр определяет, будет ли перезаписываться файл, если он уже существует. Если значение третьего входного параметра равно `false`, то в случае существования файла в каталоге назначения будет происходить ошибка и файл копироваться не будет. Если же значение третьего входного параметра равно `true`, то файл, если он существует в каталоге назначения, будет перезаписан.
- `CopyFolder «ïóòü è èìÿ êîïèðóåìîé ïàïêè», «ïóòü è èìÿ íîâîé ïàïêè», «ôëàã ïåðåçàïèñè»` — работа данного метода аналогична работе предыдущего метода, но вместо файла он копирует папку.
- `CreateFolder («ïóòü ê íîâîé ïàïêå è åå íàçâàíèå»)` — создает указанную во входном значении папку. При этом после своей работы метод возвращает ссылку на объект, указывающий на созданную папку. Данный объект, ссылка на который возвращается, поддерживает несколько методов. Метод возвращает:
  - `Path` — путь к созданной папке, если же путь не определен, то папка создана не была;
  - `Size` — размер созданной папки и ее содержимого;

- Attributes — битовую маску, указывающую, какие атрибуты определены для данной папки;
- DateCreated — дату создания папки (как время, так и день, месяц и год создания);
- DateLastAccessed — дату последнего доступа к папке (как время, так и день, месяц и год создания);
- DateLastModified — дату последнего изменения содержимого папки (как время, так и день, месяц и год создания);
- Drive — букву логического диска, на котором расположена папка;
- IsRootFolder — значение false, если папка не является корневой;
- Name — название папки. Его же можно получить использованием метода ShortName;
- ParentFolder — каталог, в котором расположена папка;
- Type — тип папки, например, может возвратить значение Ïàïêà ñ ôàéëàìè;
- ShortPath — путь к папке, включая ее имя.

■ CreateTextFile («ïóòü ê òåêñòîâîìó ôàéëó è åãî íàçâàíèå», «ôëàã ïåðåçàïèñè», «ôëàã Unicode») — создает указанный текстовый файл и возвращает объект, ссылающийся на созданный текстовый файл. Данный метод требует следующих входных значений:

- Ïóòü ê òåêñòîâîìó ôàéëó è åãî íàçâàíèå — именно указанный в данном входном значении текстовый файл и будет создан;
- Ôëàã ïåðåçàïèñè — если значение данного флага равно true, то в случае существования указанного в первом входном значении файла он будет перезаписан;
- Ôëàã Unicode — если значение флага равно true, то текстовый файл будет создан с поддержкой Unicode.

Созданный после работы данного метода объект также может использовать некоторые методы, предназначенные для работы с созданным файлом. Рассмотрим некоторые из них:

- Close — завершить работу с данным файлом;
- Column — возвращает количество столбцов, содержащихся в последней строке данного текстового файла;
- Line — возвращает количество строк, содержащихся в данном текстовом файле;
- Write «òåêñò» — записывает в данный файл указанную строку, но не переходит на следующую строку при окончании записи;
- WriteLine «òåêñò» — записывает в данный файл указанную строку и переходит на следующую строку при окончании записи;

  - `WriteBlankLines (êîëè÷åñòâî ñòðîê)` — записать в файле столько пустых строк, сколько указано во входном значении метода.

- `DeleteFile «ïóòü ê ôàéëó»` — удаляет указанный во входном значении файл.
- `DeleteFolder «ïóòü ê ôàéëó»` — удаляет указанную во входном значении папку.
- `DriveExists («áóêâà äèñêà:»)` — возвращает значение `true`, если указанный во входном параметре логический диск существует. Иначе возвращает значение `false`.
- `Drives («áóêâà äèñêà:»)` — возвращает ссылку на объект, с помощью которого можно просмотреть параметры указанного во входном значении диска. Объект, ссылка на который возвращается, поддерживает следующие методы.
  - `AvailableSpace` — количество свободного места на логическом диске в мегабайтах. Можно также воспользоваться методом `FreeSpace`.
  - `DriveLetter` — возвращает букву диска (без символа «:»).
  - `DriveType` — идентификатор типа диска. Возможны следующие идентификаторы: `0` — неизвестный тип диска; `1` — съемный; `2` — фиксированный; `3` — удаленный; `4` — компакт-диск; `5` — ОЗУ.
  - `FileSystem` — название файловой системы, установленной на диске.
  - `Path` — путь к диску.
  - `IsReady` — определяет, готов ли диск к работе. Например, если данный метод возвращает значение `false` при работе с оптическим диском, значит, в данном оптическом приводе отсутствует компакт-диск.
  - `SerialNumber` — серийный номер диска.
  - `TotalSize` — общий размер данного логического диска в мегабайтах.
  - `VolumeName` — название метки диска.
- `FileExists «ïóòü ê ôàéëó»` — возвращает значение `true`, если указанный во входном значении файл существует.
- `FolderExists «ïóòü ê ïàïêå»` — возвращает значение `true`, если указанная во входном значении папка существует.
- `GetBaseName «ïóòü ê êàòàëîãó»` — возвращает название конечного каталога в пути, указанном во входном значении.
- `GetDrive «áóêâà äèñêà»` — возвращает объект, являющийся ссылкой на указанный логический диск. После получения объекта к нему можно применить те же методы, что и к объекту, получаемому с помощью метода `Drives`. Например, можно воспользоваться методом `TotalSpace`, чтобы узнать общий размер диска.
- `GetFile «ïóòü ê ôàéëó»` — возвращает объект, являющийся ссылкой на указанный файл. После получения объекта к нему можно применить те же мето-

ды, что и к объекту, получаемому с помощью метода CreateFolder. Например, можно воспользоваться методом DateCreated, чтобы узнать дату создания файла.

- GetFolder «ïóòü ê ïàïêå» — возвращает объект, являющийся ссылкой на указанную папку. После получения объекта к нему можно применить те же методы, что и к объекту, получаемому с помощью метода CreateFolder. Например, можно воспользоваться методом DateCreated, чтобы узнать общий размер диска.
- GetFileVersion «ïóòü ê ôàéëó» — возвращает версию файла, приведенного во входном значении метода.
- GetTempName — возвращает имя последнего файла TMP, создаваемого на данном компьютере.
- MoveFile «ïóòü è èìÿ ôàéëà», «ïóòü, êóäà ôàéë áóäåò ïåðåìåùåí, à òàêæå åãî íîâîå èìÿ» — перемещает файл, указанный в первом входном параметре, туда, куда указывает второй входной параметр (и, если необходимо, меняет имя файла).
- MoveFolder «ïóòü è èìÿ êàòàëîãà», «ïóòü, êóäà êàòàëîã áóäåò ïåðåìåùåí, à òàêæå åãî íîâîå èìÿ» — перемещает каталог, указанный в первом входном параметре, туда, куда указывает второй входной параметр (и, если необходимо, меняет имя каталога).
- OpenTextFile «ïóòü è èìÿ ôàéëà», «ôëàã îòêðûòèÿ ôàéëà» — открывает файл, указанный в первом входном параметре, с доступом, указанным во втором входном параметре. Второй входной параметр может содержать следующие значения:
  - 1 — открыть файл для чтения;
  - 2 — открыть файл для перезаписи;
  - 8 — открыть файл для дозаписи (записи в конец файла, то есть, не переписывая его содержимое).

При любом способе доступа к файлу метод возвращает указатель на объект, указывающий на открытый файл. Объект, указатель на который был возвращен, поддерживает те же методы, что и объект, получаемый при вызове метода CreateTextFile. Кроме того, если файл открывается с доступом на чтение, то доступны еще и следующие методы:

- ReadLine — считать значение строки и перейти на следующую строку данного файла;
- Read(êîëè÷åñòâî ñèìâîëîâ) — считать первые n символов из строки данного файла;
- ReadAll — считать все содержимое данного файла;
- Skip — пропустить данное количество символов;

- `SkipLine` — пропустить следующую строку символов;
- `AtEndOfLine` — возвращает значение `true`, если достигнут конец строки;
- `AtEndOfStream` — возвращает значение `true`, если достигнут конец файла.

Другие возможности

Рассмотрим другие возможности, которые предоставляют объекты сервера сценариев Windows. При этом будут рассмотрены как новые объекты, так и уже описанные ранее, ведь при их описании мы не всегда знакомились со всеми доступными в них методами.

Объект WshShell

Popup («текст сообщения», «количество секунд», «текст заголовка», «тип окна»)

Метод отображает текстовое сообщение, указанное в первом входном параметре. При этом, кроме текста сообщения нужно указать следующие входные параметры.

- `Êîëè÷åñòâî ñåêóíä` — определяет количество секунд, которое вызванный диалог будет отображаться. По истечении этого времени текстовое сообщение само исчезнет.
- `Òåêñò çàãîëîâêà` — указывает текст заголовка диалога текстового сообщения.
- `Òèï îêíà` — константное выражение, определяющее количество кнопок текстового окна, а также тип выводимого окна. Возможны следующие константы (в скобках указаны числовые выражения, которым соответствуют эти константы).

Константы количества кнопок.

- `vbOkOnly` — отображать только кнопку ОК (`0`).
- `vbOkCancel` — отображает кнопки ОК и Отмена (`1`).
- `vbAbortRetryIgnore` — кнопки Прервать, Повтор и Пропустить (`2`).
- `vbYesNoCancel` — кнопки Да, Нет и Отмена (`3`).
- `vbYesNo` — кнопки Да и Нет (`4`).
- `vbRetryCancel` — кнопки Повтор и Отмена (`5`).

Константы типа окна. Они могут добавляться к одной из предыдущих констант (например, `vbOkOnly + vbCritical`).

- `vbCritical` — выводит знак ошибки (`16`).
- `vbQuestion` — знак вопроса (`32`).

- vbExclamation — знак восклицания (48).
- vbInformation — знак информации (64).

Константы кнопки по умолчанию. Они могут добавляться к одной из предыдущих констант (например, vbYesNoCancel + vbQuestion + vbDefaultButton3).

- vbDefaultButton1 — первая кнопка имеет фокус (0).
- vbDefaultButton2 — вторая кнопка имеет фокус (256).
- vbDefaultButton3 — третья кнопка имеет фокус (512).
- vbDefaultButton4 — четвертая кнопка имеет фокус (768).

Константы модальности. Они могут добавляться к одной из предыдущих констант (например, vbYesNoCancel + vbQuestion + vbDefaultButton3 + vbApplicationModal).

- vbApplicationModal — окно является модальным для текущего приложения (0).
- vbSystemModal — для всех приложений системы (4096)

ПРИМЕЧАНИЕ

Вы заметили, что входные значения некоторых методов берутся в скобки, а некоторых не берутся? На самом деле здесь все просто. Если метод возвращает значение и вы это значение получаете (то есть имеет место строка «переменная»=«метод»), то входные значения нужно брать в скобки, иначе, даже если метод возвращает значение, но вы его не принимаете (то есть имеет место строка «метод»), входные значения в скобки брать не нужно.

Метод может возвращать константу той кнопки, которую выбрал пользователь. Возможны следующие константы:

- vbOk — пользователь выбрал кнопку ОК (1);
- vbCancel — кнопку Отмена (2);
- vbAbort — кнопку Прервать (3);
- vbRetry — кнопку Повтор (4);
- vbIgnore — кнопку Пропустить (5);
- vbYes — кнопку Да (6);
- vbNo — кнопку Нет (7).

CreateShortcut ("путь к ярлыку и его имя")

Метод создает ярлык, названный в честь входного параметра. При этом следует учитывать, что указанный во входном параметре файл должен завершаться расширением LNK или URL.

Вызова данного метода еще не достаточно для создания ярлыка. Метод возвращает объект, после принятия которого именно с этим объектом и ведется дальнейшая работа. Данный объект поддерживает два свойства:

- `TargetPath` — определяет путь к файлу, на который будет создаваться ярлык;
- `Save` — после вызова этого метода ярлык будет создан.

CurrentDirectory

Свойство возвращает текущую директорию, в которой находится сценарий или которая используется в данный момент командной строкой, если сценарий вызывается из командной строки.

Рассмотрим пример работы с этим и двумя предыдущими методами. В этом примере будет создан ярлык файла, путь к которому задаст пользователь. При этом ярлык будет располагаться либо в текущем каталоге (если пользователь нажмет кнопку Да), либо в каталоге, который пользователь сам укажет.

□□□□□□□ 16.2. □□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□

```
set wshshell = WScript.CreateObject("WScript.Shell")

vibor = wshshell.Popup ( "□□□□□□□ □□□□□ □ □□□□□□□□ " & wshshell.CurrentDirectory & "?", 
"100", "□□□□□□□□ □□□□□□", vbYesNoCancel)
select case vibor
case vbYes
   set yarlik = wshshell.CreateShortcut (wshshell.CurrentDirectory & "\eto_yarlik.lnk")
   yarlik.TargetPath = "d:\aa.bmp"
    yarlik.Save
case vbNo
  path=InputBox("□□□□□□□□□□□□□□□□□□□□□□□","□□□□□□□□□□□□□□□",
wshshell.CurrentDirectory & "\eto_yarlik.lnk")
    If path <> "" Then
        set yarlik = wshshell.CreateShortcut (path)
        yarlik.TargetPath = "d:\aa.bmp"
        yarlik.Save
    Else
      MsgBox "□□ □□□ □□□□□□□..."
    End if
case vbCancel
   MsgBox "□□ □□□ □□□□□□□..."
end select
```

Environment

Метод предназначен для работы с системными переменными. Системные переменные можно посмотреть в одноименном списке диалога Переменные среды, который отобразится после нажатия кнопки Переменные среды, расположенной

на вкладке Дополнительно диалога Свойства системы. Метод также поддерживает некоторые методы, которые будут рассмотрены.

- `Count` — возвращает общее количество системных переменных, созданных на данный момент.
- `Length` — аналогичен предыдущему методу.
- `Remove(èìÿ ïåðåìåííîé)` — удаляет указанную системную переменную.
- `Item(èìÿ ïåðåìåííîé)` — возвращает значение данной системной переменной.

Рассмотрим пример работы с системными переменными. Сначала пример узнает общее количество таких переменных, потом пытается считать значение переменной `winbootdir` (как правило, такая системная переменная всегда присутствует на компьютере), а потом удаляет эту переменную и снова считывает общее количество системных переменных.

ПРИМЕЧАНИЕ

Получить значение переменной можно также с помощью подобного вызова: MsgBox wshshell.ExpandEnvironmentStrings("%systemroot%").

16.3.

```
set wshshell = WScript.CreateObject("WScript.Shell")

MsgBox wshshell.Environment.Count
MsgBox wshshell.Environment.item("winbootdir")
wshshell.Environment.Remove("winbootdir")
MsgBox wshshell.Environment.Count
```

Exec (путь и название исполняемого файла)

Метод предназначен для выполнения команд и при своем вызове возвращает объект, с помощью которого можно управлять вызванной программой. Данный объект поддерживает следующие методы.

ПРИМЕЧАНИЕ

Выполнить команду можно также с помощью метода Run. Он имеет следующий синтаксис: «переменная» = wshshell.run ("путь к программе", TRUE).

- `Terminate` — завершить вызванную программу.
- `ExitCode` — код, возвращаемый при открытии программы. Если значение этого кода равно `0`, то программа была вызвана успешно. Для этих целей можно также воспользоваться методом `Status`.
- `ProcessID` — возвращает идентификатор, присвоенный нашей вызванной программе.

Рассмотрим простой пример работы с данным методом. В этом примере вызывается Проводник, после чего отображается PID созданного нами процесса, а затем процесс завершится.

16.4.

```
set wshshell = WScript.CreateObject("WScript.Shell")

set prog = wshshell.Exec ("explorer.exe")

if prog.Status = 0 then
   MsgBox prog.ProcessID
   prog.Terminate
End if
```

SendKeys

Работа этого метода довольно интересна. Он возвращает произвольное значение после завершения работы сценария. Например, если запустить в командном процессоре (`cmd.exe`) сценарий, содержащий строку `wshshell.SendKeys "This message return over by script"`, то после завершения работы сценария в командном процессоре (то есть в строке для ввода команд командного процессора) появится указанное сообщение.

SpecialFolders

Метод предназначен для работы с пользовательскими папками. Он также поддерживает следующие методы.

- `Count` — возвращает общее количество пользовательских папок.
- `Length` — аналогичен предыдущему методу.
- `Item (èíäåêñ)` — возвращает путь к папке, определенной данным индексом. Индекс может принимать значения от `0` до общего числа папок (возвращаемое методом `Count` значение).

Рассмотрим пример, отображающий пути ко всем возможным пользовательским папкам.

16.5.

```
set wshshell = WScript.CreateObject("WScript.Shell")

For i = 0 to wshshell.SpecialFolders.Count - 1
   MsgBox wshshell.SpecialFolders.Item (i)
Next
```

Как обычно, были рассмотрены далеко не все объекты и методы, которые доступны в Windows XP. Например, был пропущен такой интересный и знаменитый объект (знаменитый потому, что в свое время именно он использовался в почтовом черве ILOVEYOU), как объект для доступа к почтовым функциям программы Outlook Express. Объект поддерживает очень много методов, не говоря уже о других объектах Windows XP, поэтому их описание могло вылиться в отдельную книгу. В любом случае, автор рассчитывал создать лишь введение в возможности сервера сценариев Windows. Если эта тема вам интересна, рекомендуется купить отдельную книгу, посвященную только ей.

Глава 17

Другие возможности

- **Вкладка Общие диалога Свойства системы**
- **Файл desktop.ini**
- **SCF-файлы**
- **Файл BOOT.INI**

Теперь кратко рассмотрим несколько возможностей настройки интерфейса оболочки Windows XP или ее конфигурации без использования реестра, команд `rundll32` и всего того, что мы с вами уже рассмотрели.

Вкладка Общие диалога Свойства системы

Существует возможность редактирования содержимого вкладки **Общие** диалога **Свойства системы** даже без доступа к реестру Windows XP. Плюсом этого метода является больше возможностей, которые с его помощью можно выполнить.

Итак, для редактирования содержимого вкладки **Общие** необходимо создать в каталоге `%systemroot%\system32` два файла — `oemINFo.ini` и `Oemlogo.bmp`. Второй файл просто является картинкой, которая будет добавлена на вкладку **Общие**, а пример содержимого файла `oemINFo.ini` рассмотрим в листинге 17.1.

□□□□□□□ **17.1.** □□□□□□□ □□□□□□ oemINFo.ini

```
[Support Information]
Line1="□□ □□□□□□ □□□□□□□"
Line2="□□□□□□□□□□□□□□□□□□□□□□□□□□□□..."
Line3="...□ □□□□□ □□□□ :-)"

[General]
Manufacturer = "Parad0x-DeS1gn"
Model = "□□□□□□ Pentium □ Celeron"
```

Назначение ключевых слов данного INI-файла легко понять на примере того, что он делает. Посмотрим на рис. 17.1.

Рисунок, отображаемый слева внизу, является файлом `oemlogo.bmp`, тогда как кнопка **Сведения о поддержке** создается при помощи блока `[Support Information]` файла `oemINFo.ini`, а текст перед названием процессора — при помощи блока `[General]` файла `oemINFo.ini`.

Файл desktop.ini

Еще один интересный специальный файл, с помощью которого можно выполнить настройку оболочки Windows XP. Например, с его помощью можно изменить значок для папки, в которой он будет находиться, создать для нее описание и сделать многое другое. Для примера попробуем изменить изображение для отдельной папки и создать для нее описание. Для этого необходимо сделать следующее.

1. Создать в папке файл с названием `desktop.ini` (желательно также скрыть его).
2. Присвоить папке атрибут системной. Это выполняется с помощью команды `attrib +S «ïóòü ê ïàïêå»`.

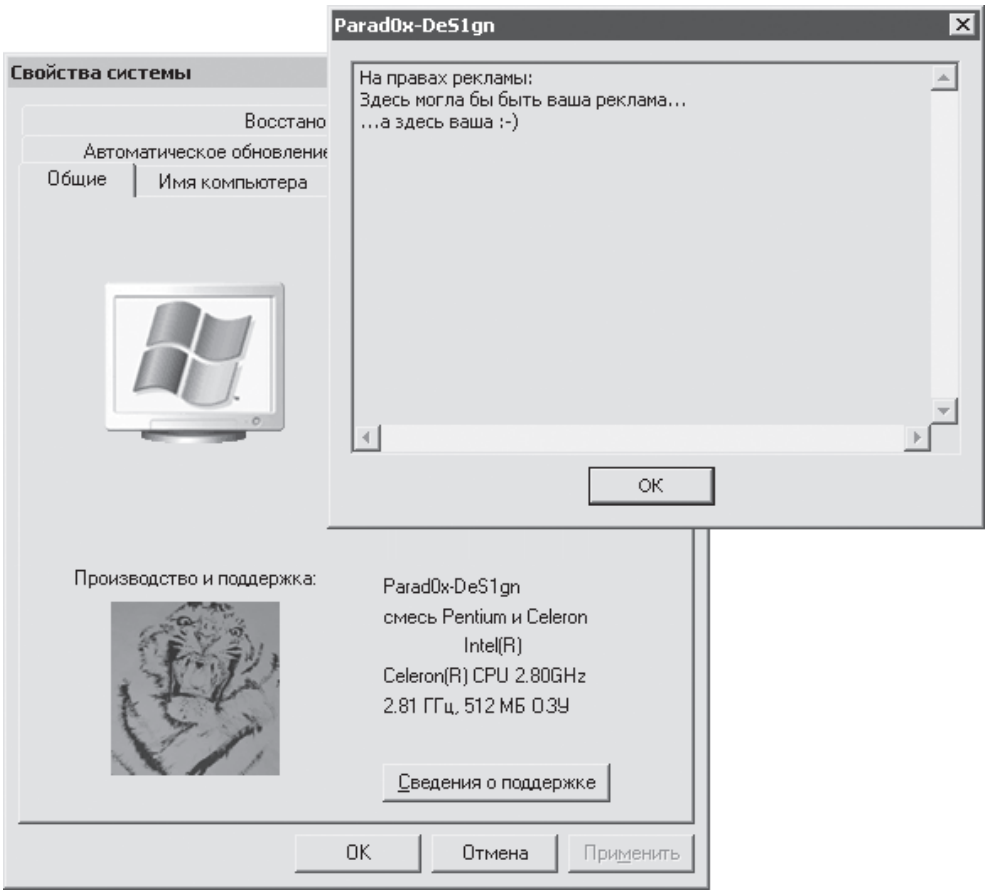


Рис. 17.1. Результат применения файлов oemINFo.ini и Oemlogo.bmp

Если вы уже сделали папку системной, а также создали в ней файл `desktop` с расширением `ini`, то приведем в листинге 17.2 небольшой пример содержимого файла `desktop.ini`.

17.2. desktop.ini

```
[.ShellClassInfo]
IconFile = c:\windows\system32\shell32.dll
IconIndex = 34
InfoTip="
"
```

Назначение ключевых слов данного файла можно понять при взгляде на рис. 17.2.

Ключевое слово `IconFile` указывает путь к библиотеке, содержащей необходимый вам рисунок папки, а ключевое слово `IconIndex` определяет индекс необхо-

димого вам изображения в данной библиотеке. Ключевое слово `InfoTip` определяет подсказку, отображаемую для данного файла.

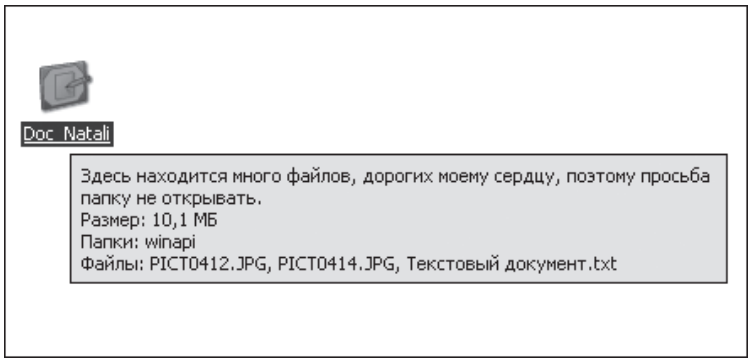


Рис. 17.2. Использование файла desktop.ini для изменения изображения папки

С помощью файла `desktop.ini` можно также запретить создание в папке других папок и файлов (при этом есть возможность во вложенных папках создать папки и файлы). Для этого необходимо в блоке `[.ShellClassInfo]` создать следующие строки:

```
UICLSID={7BD29E00-76C1-11CF-9DD0-00A0C9034933}
CLSID={FF393560-C2A7-11CF-BFF4-444553540000}
```

После этого рисунок папки изменится, а при попытке записи в папку будет выдаваться сообщение (рис. 17.3). Если же необходимо, чтобы рисунок папки не менялся, то строку `CLSID={FF393560-C2A7-11CF-BFF4-444553540000}` нужно удалить.

Блок `[.ShellClassInfo]` может иметь ключевое слово `LocalizedResourceName`, которое переопределяет название папки. Например, если в файле `desktop.ini` для нашей папки создать строку `LocalizedResourceName=@shell32.dll,-21765`, то название папки `Doc_Natali` изменится на название `Application Data`. При этом, к сожалению, ключевое слово `LocalizedResourceName` не поддерживает прямой текст. Другими словами, нужно обязательно указывать текст, хранящийся в какой-либо библиотеке.

Но, кроме названия самой папки, с помощью файла `desktop.ini` можно изменить названия файлов, содержащихся в этой папке. Для этого служит блок INI-файла `[LocalizedFileNames]`. Он может включать в себя строки формата «`èñ-òèíîå íàçâàíèå ôàéëà`»=«`áèáëèîòåêà, è èíäåêñ íîâîãî íàçâàíèÿ â íåé`». К сожалению, новые названия прямым текстом указывать нельзя.

Например, если в папке содержится файл `PICT0412.JPG`, то для изменения названия этого файла на `Ðàáî÷èé ñòîë (ñîçäàòü ÿðëûê)`, нужно воспользоваться строкой `PICT0412.JPG=@sendmail.dll,-21`.

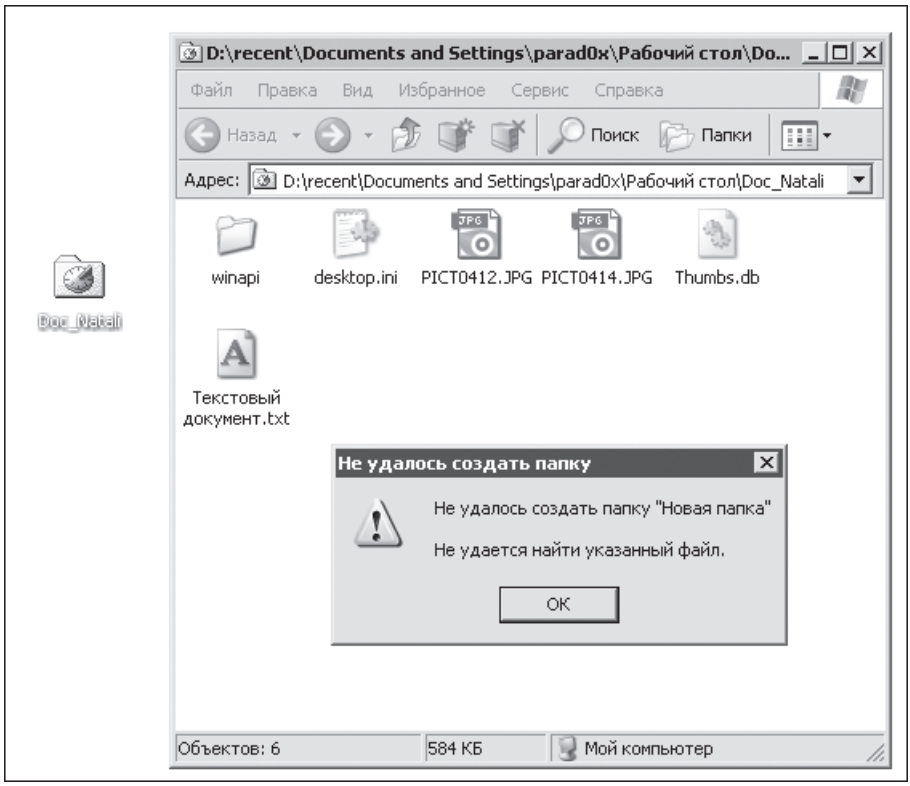


Рис. 17.3. Ошибка при создании папки или файла

SCF-файлы

Файлы с таким расширением являются командными файлами оболочки Windows и используются для различных целей. Например, можно создать файл, который будет сворачивать все окна. Его содержимое приведено в листинге 17.3. Если ввести подобный текст в текстовый файл, а потом присвоить ему расширение SCF, то будет создан соответствующий файл (расширение файла будет скрыто), выполнение которого свернет все окна.

ПРИМЕЧАНИЕ

При этом значок, используемый файлом, можно изменять.

Листинг 17.3.

```
[Shell]
Command=2
IconFile=explorer.exe,3
```

```
[Taskbar]
Command=ToggleDesktop
```

Можно также создать файл для просмотра каналов Интернета. Для этого используется представленный в листинге 17.4 текст.

Листинг 17.4.

```
[Shell]
Command=3
IconFile=shdocvw.dll,-118

[IE]
Command=Channels
```

Можно создать файл для открытия Проводника Windows XP, как показано в листинге 17.5.

Листинг 17.5.

```
[Shell]
Command=2
IconFile=explorer.exe,1

[Taskbar]
Command=Explorer
```

Вообще, это довольно интересные файлы, так как они могут хранить любой другой посторонний текст, который не будет влиять на функциональность данных файлов. При этом никому и в голову не придет посмотреть содержимое этих файлов на предмет посторонних записей, поэтому в них можно запросто хранить конфиденциальную информацию.

Файл BOOT.INI

Данный файл используется при установке нескольких операционных систем семейства Windows на одном компьютере для реализации возможности выбора между ними при загрузке системы. Рассмотрим содержимое этого файла. Файл расположен в каталоге `%systemdrive%`, хотя наиболее удобным способом его редактирования является стандартная программа `msconfig`. Она имеет вкладку BOOT.INI, с помощью которой довольно легко работать с этим файлом, если знать его синтаксис. По этой причине сначала рассмотрим синтаксис данного файла.

Файл `BOOT.INI` состоит из двух блоков. Первый из них называется `[boot loader]` и содержит информацию о времени ожидания выбора пользователем операционной системы, а также об операционной системе, загрузка которой будет выполнена в случае, если пользователь так и не выбрал за предоставленное время конкретную операционную систему. Второй блок называется `[operating systems]`.

В нем находится информация обо всех операционных системах, которые можно будет выбрать для загрузки с помощью списка операционных систем. Посмотрим на листинг 17.6, в котором представлен пример файла `BOOT.INI`.

17.6. BOOT.INI

```
[boot loader]
timeout=5
default=multi(0)disk(0)rdisk(0)partition(2)\WINDOWS

[operating systems]
multi(0)disk(0)rdisk(0)partition(3)\WINDOWS="Windows XP Professional Test" /
noexecute=optin /fastdetect
multi(0)disk(0)rdisk(0)partition(2)\WINDOWS="Windows XP Tablet PC Edition" /
noexecute=optin /fastdetect
multi(0)disk(0)rdisk(0)partition(1)\WINDOWS="Windows XP Professional Recent" /
fastdetect /NoExecute=OptIn
C:\CMDCONS\BOOTSECT.DAT="Windows XP Recovery Console" /cmdcons
```

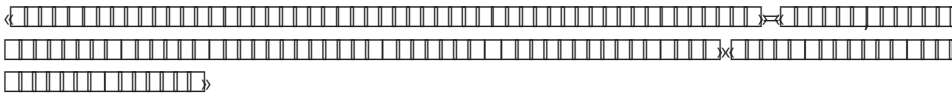
Boot loader

Блок `[boot loader]` содержит всего два ключевых слова.

- `Timeout` — определяет количество секунд, которые система будет ожидать выбора пользователем загружаемой операционной системы перед тем, как загрузить операционную систему, указанную в ключевом слове `default`.
- `Default` — указывает путь к загружаемой по умолчанию операционной системе (к системному каталогу данной операционной системы). Иными словами, данный путь должен соответствовать одному из путей, указанному в блоке `[operating systems]` перед символом `=`.

Operating systems

Блок `[operating systems]` содержит столько строк, сколько операционных систем установлено на компьютере. При этом каждая строка имеет такой формат:



Как можно заметить из листинга, путь к операционной системе строится весьма необычным образом — не с помощью буквы логического диска, на котором установлена операционная система, а с помощью номера партиции, в которой установлена операционная система, а также номера диска. Если вы не понимаете ничего в таком способе построения пути, то просто можете посмотреть, с помощью какого пути загружалась данная операционная система. Для этого предназначен па-

раметр строкового типа SystemBootDevice, расположенный в ветви реестра HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control.

ARC-имена состоят из четырех частей, совместно описывающих букву диска, на котором нужно искать операционную систему. Вкратце рассмотрим эти части.

- Первая определяет контроллер для данного жесткого диска и может быть равна либо multi(«íîìåð êîíòðîëëåðà äèñêà»), либо scsi(«íîìåð scsi-àäàïòåðà â ñèñòåìå»). При этом следует учитывать, что нумерация контроллеров диска начинается с нуля, то есть для первого контроллера диска данная часть будет равно multi(0).
- Вторая определяет номер диска, подключенного к указанному в первой части контроллеру. Для SCSI-дисков вторая часть равна disk(«íîìåð äèñêà, íà÷èíàÿ ñ íóëÿ»). Если же первая часть равна multi(«íîìåð êîíòðîëëåðà äèñêà»), то вторая часть всегда должна быть равна disk(0) — она не используется и будет игнорироваться.
- Третья определяет номер диска, подключенного к указанному в первой части контроллеру. Для SCSI-дисков третья часть всегда равна rdisk(0), так как она не используется, ведь номер диска был задан ранее. Если же первая часть равна multi(«íîìåð êîíòðîëëåðà äèñêà»), то третья часть должна быть равна disk(«íîìåð äèñêà, íà÷èíàÿ ñ íóëÿ»).
- Четвертая указывает конкретный раздел на жестком диске, в котором содержатся файлы операционной системы, и равна partition(«íîìåð ðàçäåëà, íà÷èíàÿ ñ åäèíèöû»).

Можно также посмотреть ключи, которые использовались при запуске операционной системы. Для этого предназначен параметр строкового типа SystemStartOptions, расположенный в ветви системного реестра HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control. Теперь поговорим о ключах. Что же они делают?

- /BASEVIDEO — говорит системе, что данная операционная система должна быть загружена с использованием стандартного графического драйвера vga.sys. Это может понадобиться в случае, когда недавно установленный в операционной системе графический драйвер ведет себя некорректно.
- /BAUDRATE — определяет скорость передачи в бодах, которая будет использоваться при отладке запуска операционной системы по определенному COM-порту. Данный ключ должен использоваться только вместе с ключом /DEBUG. По умолчанию используется скорость 9600 бод для модемного соединения и 19 200 бод для нуль-модемного кабеля.
- /BOOTLOG — установка данного ключа приводит к созданию файла журнала (файл NTBTLOG.TXT, расположенный в каталоге %systemroot%) при каждой загрузке операционной системы. В данный файл журнала будет помещаться информация обо всех загруженных при запуске Windows XP драйверах, а также о тех драйверах, загрузить которые не удалось.

- /CRASHDEBUG — говорит системе, что при запуске данной операционной системы необходимо также запускать отладчик ядра в состоянии ожидания. Это может понадобиться при возникновении аварийной остановки при загрузке операционной системы (чтобы понять причину возникновения экрана BSOD).
- /DEBUG — при запуске операционной системы также должен быть запущен отладчик, доступ к которому можно получить по COM-порту удаленного компьютера. Данный режим может быть полезен при возникновении аварийной остановки при запуске операционной системы.
- /DEBUGPORT=«COM-ïîðò» — определить COM-порт, по которому будет вестись работа с отладчиком при возникновении аварийной остановки. По умолчанию используется порт COM1.
- /FASTDETECT — говорит о том, что программа ntdetect.com не должна определять установленные на компьютере устройства. Вместо нее это будет делать система Plug and Play.
- /MAXMEM=«êîëè÷åñòâî ìåãàáàéò îïåðàòèâíîé ïàìÿòè» — определяет количество оперативной памяти, которое будет использовать данная операционная система во время своей работы.
- /NODEBUG — не выводить отладочную информацию на экран компьютера.
- /NOGUIBOOT — не отображать графическую заставку Windows.
- /NOSERIALMICE=«COM-ïîðò» — запретить определение мыши на указанных COM-портах (COM-порты можно писать через запятую), если она там подключена.
- /SAFEBOOT:«ðåæèì» — говорит системе, что данная операционная система должна загружаться в одном из безопасных режимов. При этом доступны следующие режимы:
  - MINIMAL — обычный безопасный режим;
  - MINIMAL(ALTERNATESHELL) — безопасный режим с поддержкой командной строки;
  - NETWORK — безопасный режим с загрузкой сетевых драйверов;
  - DSREPAIR — безопасный режим для восстановления каталога Active Directory.
- /SOS — отображать при загрузке операционной системы имена всех запускаемых драйверов.

MSconfig.exe

С помощью вкладки BOOT.INI все приведенные ключи загрузки можно автоматически добавить к выделенной в данный момент строке операционной системы, не беспокоясь об их корректности. Для этого необходимо сначала выделить строку запуска операционной системы, а потом установить необходимые флажки напротив часто используемых ключей, отображенных на вкладке BOOT.INI программы msconfig. Если какого-то необходимого ключа в данном списке вы не найдете,

то можно нажать кнопку Дополнительно, после чего перед вами отобразится список дополнительных ключей.

И наконец, можно нажать кнопку Проверить все пути загрузки, чтобы определить, соответствуют ли новые пути загрузки операционных систем (если вы их изменяли) реальному их расположению на жестком диске компьютера.

Bootcfg.exe

Это еще одна программа (программа командной строки) для работы с файлом `BOOT.INI`. Причем данная программа имеет один большой плюс — с ее помощью можно редактировать файл `BOOT.INI` удаленной операционной системы. Мы не будем вдаваться в подробности работы этой программы, так как работа с ней описана в Центре справки и поддержки.

Глава 18

Стандартные каталоги Windows и их содержимое

Еще одной темой, которая может вас заинтересовать, является содержимое каталогов Windows. Согласитесь, ведь каждый иногда задавал себе такие вопросы: «А зачем нужен этот каталог?» «А зачем нужно столько много каталогов?» «А зачем нужны эти файлы?» Наконец, попробуем разобрать назначение если не всех стандартных каталогов Windows и файлов в них, то многих из них.

- `%systemroot%\$íàçâàíèå îáíîâëåíèÿ$` — папки создаются устанавливаемыми на компьютер пакетами обновлений Windows XP и предназначены для возможности удаления установленного обновления (откат к предыдущему состоянию). Как правило, внутри таких папок есть программа, удаляющая обновления, но можно воспользоваться и диалогом Установка и удаление программ.
- `%systemroot%\CURSORS` — хранит доступные на компьютере файлы указателей. Чтобы можно было установить свой указатель, нужно созданный файл указателя поместить в эту папку.
- `%systemroot%\FONTS` — содержит все установленные на компьютере шрифты. Чтобы установить шрифт, нужно в меню Файл данной папки выбрать команду Установить шрифт.
- `%systemroot%\Inf` — включает в себя все установленные на компьютере INF-файлы. Данная папка скрыта.
- `%systemroot%\Installer` — хранит большинство из пакетов установщиков Windows, которые вы когда-либо инсталлировали на компьютер. Если вам необходимо установить или переустановить какую-нибудь программу, инсталлируемую при помощи MSI-пакетов, и при этом вы не имеете ее установочного диска, но раньше уже устанавливали эту программу, то можно попытаться найти соответствующий установочный файл в данной папке. По умолчанию эта папка скрыта, а пакеты, которые в ней содержатся, имеют числовые номера (а не реальные номера, которые им были даны), поэтому самым простым способом определить принадлежность пакета является вкладка Сводка диалога Свойства пакета.
- `%systemroot%\LastGood` — если в системе присутствует эта папка, то она может хранить копии различных файлов. Сейчас администраторы многих компаний удаляют стандартные игры с пользовательских компьютеров. Но даже после их удаления копии могут находиться в нескольких различных местах файловой системы Windows XP. Например, в этой папке или в папке, используемой для хранения точек восстановления программой Восстановление системы, и папке, используемой службой индексации.
- `%systemroot%\ntds` — содержит файлы базы данных и журналы каталога, используемые контроллером домена Active Directory. Данный каталог является критически важным для работы контроллера домена.
- `%systemroot%\sysvol` — используется для тиражирования файлов между контроллерами доменов Active Directory. Он также используется службой индексирования.
- `%systemroot%\repair` — хранит архивные копии файлов кустов реестра, создаваемых ASR при работе программы архивации `ntbackup.exe` и используемых

ею при восстановлении системы. Данный каталог также хранит некоторые файлы, не относящиеся к кустам реестра.

- `autoexec.nt` — используется по умолчанию для инициализации среды MS-DOS при восстановлении системы.
- `config.nt` — применяется по умолчанию для инициализации среды MS-DOS при восстановлении системы.
- `setup.log` — хранит перечень всех файлов, которые были установлены при установке операционной системы (описывается только содержимое каталога Windows и его подкаталогов).
- `secDC.inf` — включает в себя параметры безопасности для доменных контроллеров, которые будут к ним применены после восстановления системы.
- `secSetup.inf` — содержит параметры безопасности по умолчанию, которые будут применены после восстановления системы.
- `smss.ASR` — является обычной программой `smss.exe`, расширение которой было изменено. `Smss.exe` — диспетчер сеансов, начинающий работу при входе пользователя в систему и управляющий запуском различных сервисов и служб.
- `NTDLL.ASR` — является копией библиотеки `ntdll.dll`, расширение которой было изменено.

В этом каталоге также содержатся файлы кустов `DEFAULT`, `SAM`, `SECURITY`, `SOFTWARE`, `SYSTEM`. Их описание будет приведено при рассказе о содержимом каталога `%systemroot%\system32\config`.

- `%systemroot%\system32\CatRoot` — хранит все цифровые подписи драйверов и системных файлов операционной системы Windows (в виде файлов с расширением CAT). Все они хранятся в одном из вложенных разделов данного каталога, название которого создано в формате CLSID-номера ActiveX-объекта (GUID-формате). Кроме цифровых подписей для стандартных драйверов и системных файлов, в этом каталоге также могут находиться цифровые подписи для драйверов и других файлов устройств, которые были протестированы в лаборатории Microsoft и помещены в список HCL (список совместимых аппаратных средств). После успешного тестирования устройства Microsoft передает производителю этого устройства файл каталога (CAT-файл) для его продукции, который впоследствии должен поставляться вместе с каждым экземпляром данного устройства. Именно наличие этого файла каталога и проверяется при установке устройства, если в диалоге Параметры подписывания драйверов установлен переключатель Предупреждать — каждый раз предлагать выбор действия или переключатель Блокировать — запретить установку неподписанного драйвера программного обеспечения.
- `%systemroot%\system32\config` — является, наверное, самым важным каталогом в Windows — именно в нем находятся главные файлы кустов реестра.

Кроме них, каталог хранит и другие файлы, к реестру не имеющие никакого отношения. Вкратце рассмотрим его содержимое.

- `AppEvent.Evt` — является журналом событий приложений. Именно с ним мы и работали при изучении оснастки `eventwvr.msc` (Просмотр событий (локальных) ▸ Приложение).
- `DEFAULT` — содержит раздел реестра `HKEY_USERS\.DEFAULT`.

ПРИМЕЧАНИЕ

Кроме файла DEFAULT, в описанном выше каталоге находятся также файлы DEFAULT.LOG и DEFAULT.SAV. Первый из них является журналом транзакций и содержит все изменения параметров данного куста за текущий сеанс работы компьютера. Второй же включает в себя куст HKEY_USERS\.DEFAULT, который был создан во время тестовой фазы установки операционной системы. То же относится и к другим файлам кустов — все они имеют своих тезок с расширениями SAV и LOG.

- `DnsEvent.Evt` — является журналом событий сервера DNS. В этой книге не рассматривалась возможность работы с журналом сервера DNS, поскольку это не являлось главной темой книги.
- `FileRep.Evt` — прдставляет собой первый журнал событий репликации.
- `NTDS.Evt` — является журналом событий службы каталогов Windows XP.
- `NtFrs.Evt` — представляет собой второй журнал событий репликации.
- `SAM` — является разделом реестра `HKEY_LOCAL_MACHINE\SAM`. По умолчанию данный раздел не может просмотреть даже администратор, хотя это может сделать система. Да вам и самим доступно это проверить — просто приспособьте приведенный ранее метод получения прав администратора, если у вас есть права опытного пользователя, не к взлому, а к получению окна редактора реестра, открытого от имени учетной записи системы.
- `SecEvent.Evt` — представляет собой журнал событий безопасности. Именно с ним мы и работали при изучении оснастки `eventwvr.msc` (Просмотр событий (локальных) ▸ Безопасность).
- `SECURITY` — является разделом реестра `HKEY_LOCAL_MACHINE\SECURITY`. По умолчанию данный раздел не может просмотреть даже администратор, хотя это может сделать система. Содержимое этой ветви вы также сможете просмотреть, модернизировав метод получения прав администратора, если у вас есть права пользователя.
- `SOFTWARE` — содержит раздел реестра `HKEY_LOCAL_MACHINE\SOFTWARE`.
- `SysEvent.Evt` — является журналом системных событий. Именно с ним мы работали при изучении оснастки `eventvwr.msc` (Просмотр событий (локальных) ▸ Система).
- `System` — содержит раздел реестра `HKEY_LOCAL_MACHINE\SYSTEM`.

  - `Userdiff` — используется для обновления профилей пользователей более ранних версий операционных систем с целью их применения в Windows NT 4.0 и в более новых операционных системах Windows.

- `%systemroot%\SYSTEM32\dhcp\Backup` — содержит резервную копию базы данных этого DHCP-сервера (DHCP-сервер предназначен для выдачи временных IP-адресов и автоматического конфигурирования стека TCP/IP на компьютерах сети). В резервную копию входят следующие данные:
  - сведения о выданных IP-адресах и времени окончания их аренды;
  - настройки параметров данного DHCP (параметрами корпорация Microsoft назвала дополнительные сведения о конфигурировании стека TCP/IP, которые получают компьютеры вместе с IP-адресом);
  - сведения об областях действия (перечень компьютеров, которые могут воспользоваться услугами данного DHCP-сервера).

  Интервал обновления резервной копии базы данных DHCP-сервера по умолчанию равен 60 минутам, но его можно переопределить с помощью `DWORD`-параметра `BackupInterval` (значение указывается в минутах) из ветви реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\DhcpServer\Parameters`.

- `%systemroot%\system32\DllCache` — содержит копии всех системных файлов Windows XP (не только библиотек, но и программ) и необходим при повреждении или несанкционированном системой изменении оригинального системного файла для его переустановки. Данный каталог сжат и скрыт.

- `%systemroot%\system32\DNS` — хранит файлы настройки DNS (служба для трансляции символьных адресов компьютеров (например, www.mail.ru) в числовые IP-адреса (например, 192.100.1.34) или, если используется зона обратного разрешения, наоборот), а также сами файлы зон данного DNS. Файлы зон являются обычными текстовыми файлами с расширением DNS, которые содержат ресурсные записи (существует много типов ресурсных записей, но наиболее используемыми являются записи типа A, которые как раз и предназначены для трансляции символьных адресов в числовые) только о компьютерах данной зоны DNS.

  Следует учитывать, что способ хранения зон в виде текстовых файлов считается устаревшим. Если в сети используется Active Directory, то предпочтительнее хранить зоны в виде объектов объекта контейнерного типа `dnsZone`, что дает определенные преимущества.

- `%systemroot%\system32\IAS` — содержит две очень интересные системные базы данных, описывающие те или иные разрешения для протоколов, сервисов, а также сведения о различных используемых в системе свойствах.

- `%systemroot%\system32\restore` — хранит файлы программы Восстановление системы, а также саму эту программу (`rstrui.exe`). Одним из наиболее интересных файлов этого каталога является файл `filelist.xml`, описывающий файлы, каталоги, а также расширения файлов, которые не должны входить в точку восстановления или, наоборот, должны входить в точку восстановления.

- `%systemroot%\system32\wins` — содержит базу данных сервера WINS. WINS — это сетевая служба, предназначенная для трансляции MAC-адресов компьютеров в NetBios-имена. Если для трансляции имен используется WINS-сервер, то WINS-клиент, установленный на каждом компьютере по умолчанию, при включении компьютера отправляет сообщение WINS-серверу с просьбой выделить ему NetBios-имя для возможности работы в сети. После этого WINS-сервер предлагает компьютеру доступное имя, и, если это имя подходит компьютеру, он его принимает (WINS-клиент может послать несколько сообщений на выделение имени сразу нескольким WINS-серверам, установленным в сети, в этом случае WINS-клиент будет использовать первое полученное имя, а остальные отвергнет).

Приложение 1. Библиотеки Windows

Приложение описывает некоторые из стандартных библиотек Windows, расположенных в каталоге `%systemroot%\system32`.

- `HAL.DLL` — представляет собой уровень HAL (уровень абстракции оборудования). Этот уровень является как бы посредником между оборудованием компьютера и операционной системой (так же раньше характеризовали функции операционной системы, только она была посредником между оборудованием и программами). Он представляет общий интерфейс для взаимодействия с классами оборудования. Другими словами, уровень HAL позволяет программисту не забивать себе голову такими вопросами, как возможные производители и серии, например, видеокарт, которые будут поддерживать его программу. Вместо этого программист общается с классом, характеризующим настройки любой видеокарты. В дальнейшем данный класс с помощью уровня HAL будет преобразован в команды конкретной видеокарты, установленной на компьютере пользователя.

  Библиотека `HAL.DLL` загружается в памяти ядра при запуске операционной системы.

- `Msgina.dll` — компонент графической идентификации и аутентификации. Именно эта библиотека по умолчанию (ее можно переопределить с помощью параметра `GinaDll` ветви реестра `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon`) используется при передаче пароля и логина пользователя программе `lsass.exe` в момент, когда пользователь регистрируется в системе.

- `Msi.dll` — используется службой Windows Installer и является ее составной частью. По версии данной библиотеки (вкладка Версия диалога Свойства библиотеки) можно определить, обновлялся ли ее файл или используется стандартная библиотека Windows. Если файл библиотеки имеет версию, оканчивающуюся не нулем, то он является обновленным. Если же версия файла библиотеки оканчивается нулем, то библиотека является стандартной, поставляемой с Windows, и не изменялась.

  Плюсом службы Windows Installer является не только возможность взаимодействия с Active Directory, совмещения нескольких устанавливаемых компонентов или программ в одном пакете MSI, а также установка программ с административными правами, независимо от прав пользователя, но и возможность выполнения отката, если установка была прервана непредвиденной перезагрузкой, ошибкой или другими действиями пользователя.

- `Msv1_0.dll` — функции данной библиотеки используются для обеспечения проверки локальной безопасности компьютера: это определение прав пользователей, защита секретных объектов и объектов доверенных доменов. Для своей работы функции используют ветвь реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Lsa`.
- `Newdev.dll` — используется операционной системой для установки нового Plug and Play-оборудования. Вкратце это можно описать следующим образом. После подключения нового оборудования к компьютеру менеджер Plug and Play запрашивает у драйвера шины, к которой было подключено устройство, информацию об устройстве — производит энумерацию подключенного устройства. После этого менеджер Plug and Play запускает библиотеку `newdev.dll`, которая и вызывает функции установки нового устройства. Эта библиотека создает список драйверов, совместимых с новым устройством, выбирает лучший из них и запускает мастер обнаружения нового устройства. Дальше управление установкой нового устройства передается программе `Setup.exe`.

ПРИМЕЧАНИЕ

Информация о списке драйверов и драйвере, который был выбран, заносится (при установке нового оборудования) в файл журнала, путь и имя которого можно посмотреть или определить в параметре строкового типа LogPath ветви реестра HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Setup.

- `powrprof.dll` — предназначена для управления электропитанием компьютера — именно ее функции применяются при изменении настроек электропитания с помощью соответствующего апплета (`powercfg.cpl`).
- `setupapi.dll` — при описании библиотеки `newdev.dll` говорилось, что она вызывает функции установки нового устройства. Одной из таких функций является чтение содержимого INF-файлов, поставляемых вместе с устройством. Именно для этой операции и применяется библиотека `setupapi.dll`.
- `umpnpmgr.dll` — является диспетчером Plug and Play режима пользователя (то есть хранит WinAPI-функции для работы Plug and Play). При описании библиотеки `newdev.dll` диспетчер Plug and Play для простоты не разделялся на диспетчер режима ядра и диспетчер режима пользователя, однако на самом деле вначале новое устройство работает с диспетчером режима ядра, который после сбора всей необходимой информации передает управление диспетчеру режима пользователя. Он, в свою очередь, передает управление библиотеке `newdev.dll`.

Приложение 2. Параметры различных программ

Приложение содержит краткий список наиболее интересных параметров различных программ, входящих в поставку Windows XP.

Control.exe

Программа предназначена специально для открытия значка панели управления и может вызываться со следующими параметрами.

- `USERPASSWORDS2` — вызвать диалог изменения автоматического входа пользователя в систему.
- `PRINTERS` — вызвать папку Принтеры.
- `fonts` — Шрифты.
- `admintools` — Администрирование.
- `SCHEDTASKS` — Назначенные задания.
- `NETCONNECTIONS` — Сетевые подключения.
- `SCANNERCAMERA` — вызвать папку Сканеры и камеры, но если она не существует, тогда открывается корневой диск.
- `infrared` — вызвать апплет Инфракрасная связь.
- `international` — Язык и региональные создания.
- `telephony` — Телефон и модем.
- `keyboard` — Клавиатура.
- `mouse` — Мышь.
- `ports` — вызвать апплет Свойства системы, открытый на вкладке Имя компьютера.
- `date/time` — Дата и время.
- `color` — вызвать апплет Свойства: Экран, открытый на вкладке Оформление.
- `desktop` — Свойства: Экран.
- `sticpl.cpl` — открыть папку Сканеры и камеры (данная команда работает не всегда, но, как правило, после второго вызова команды папка отображается).

Wab.exe

Программа является диалогом Адресная книга почтовой программы Outlook Express, но может вызывать и другие диалоги.

- /Find — Поиск людей.
- /Open — диалог выбора файла адресной книги.
- /New — диалог создания файла адресной книги.
- /ShowExisting — вызвать диалог Адресная книга и показать существующие адресные книги.
- /Certificate — диалог работы с сертификатами (если такая возможность присутствует в системе).
- /All — показать все адресные книги.
- /? — вызвать список возможных команд.

Wabmig.exe

Программа является диалогом Импорт адресной книги, но имеет несколько дополнительных возможностей.

- IMPORT — вызвать диалог Импорт адресной книги.
- EXPORT — вызвать диалог Экспорт адресной книги.

Msimn.exe

После описания стольких программ, входящих в поставку почтовой программы Outlook Express, наверное, пора рассказать и о ее параметрах.

- /mailurl:«e-mail àäðåñ» — вызвать окно создания сообщения.
- /outnews — открыть программу чтения новостей Outlook.
- /newsurl:«àäðåñ» — открыть новостной сервер.
- /nws:«àäðåñ» — открыть новостной сервер.
- /eml:«àäðåñ» — открыть сервер.

Iexplore.exe

Программа является стандартным браузером Windows — Internet Explorer — и может вызываться со следующими параметрами.

- -nohome — не открывать стартовую страницу.
- -eval — заставить браузер работать в режиме совместимости.
- -Embedding — открывает браузер Internet Explorer во «встроенном» режиме (то есть когда окно браузера не отображается — создается лишь его процесс).

Explorer.exe

Всем известна эта программа стандартной оболочки Windows, служащая еще и в качестве Проводника. Она также имеет несколько интересных параметров, которые могут вам понадобиться. Один из них /select, «ïóòü ê êàòàëîãó». Вызов

Проводника с использованием данного параметра в некоторых случаях может быть очень полезен — он открывает каталог, предыдущий указанному, и выделяет указанный последним каталог. Например, если вам необходимо отредактировать настройки безопасности для папки, скажем, config, находящейся в папке %systemroot%\system32, то достаточно будет ввести команду explorer /select, %systemroot%\system32\config. После этого система откроет папку %systemroot%\system32 и сразу же выделит в ней папку config (и вам не придется тратить время на ее поиск).

Nusrmgr.cpl

Апплет предназначен для работы с учетными записями пользователей и может обрабатывать такие параметры.

- ,initialTask=ChangePicture — открыть страницу изменения рисунка, идентифицирующего учетную запись текущего пользователя.
- ,initialTask=ChangePassword — открыть страницу изменения пароля для учетной записи текущего пользователя.
- ,initialTask=ChangeName — открыть страницу изменения имени для учетной записи текущего пользователя.
- ,initialTask=ChangePassport — открыть страницу создания паспорта .NET для учетной записи текущего пользователя.
- ,initialTask=ChangeType — открыть страницу изменения типа учетной записи текущего пользователя.

Setup.exe

Программа является мастером установки Windows, хотя с ее помощью можно не только устанавливать операционную систему. Благодаря параметру -asrquicktest можно восстановить основную конфигурацию Windows с помощью информации, созданной при помощи ASR (функция из программы ntbackup.exe).

Icwconn1.exe

Программа является мастером подключения к Интернету (позволяет подключиться к Интернету с использованием новой учетной записи или уже существующей) и может использовать для своей работы следующие параметры.

- /checkoemcustini — запустить мастер установки с использованием конфигурации OEM.
- /smartstart — выполнить быструю настройку (при этом после данного параметра должно идти одно из ключевых слов: lan (подключение по сети), manual (вручную настроить подключение), auto (автоматически настроить подключение), new (новое подключение)).
- /skipintro — пропустить первый шаг мастера.

Unregmp2.exe

Программа применяется для установки и удаления программы Проигрыватель Windows Media и находится в каталоге `%systemroot%\INF`. С ее помощью можно выполнить некоторые интересные операции.

ВНИМАНИЕ

Желательно не использовать эту программу без особой нужды — только в случае неработоспособности Проигрывателя Windows Media. К тому же следует учитывать возможность замены параметров реестра, создаваемых различными дополнительными установленными кодеками. А вследствие этого — переустановку кодеков после использования параметров этой команды.

- `/HideWMP /SetShowState` — удалить из реестра многие сведения о расширениях и ActiveX-объектах Проигрывателя Windows Media, а также удалить ярлык проигрывателя из списка Программы меню Пуск.
- `/ShowWMP /SetShowState` — установить по умолчанию все удаленные предыдущей командой ветви реестра, а также восстановить ярлык проигрывателя в списке Программы меню Пуск. Другим способом восстановления ярлыка проигрывателя является использование параметра `/Shortcuts` — он также пересоздает параметры реестра, определяющие настройки создания различных ярлыков проигрывателя.
- `/SetWMPAsDefault` — переустановить по умолчанию настройки расширений файлов, используемых Проигрывателем Windows Media, и другие настройки реестра.
- `/AddNewExtensions` — в основном эквивалентен предыдущему, но восстанавливает настройки большего числа параметров реестра, предназначенных для взаимодействия с музыкальными файлами различных расширений.
- `/ISVInstall` — в основном восстанавливает настройки из ветви реестра `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Multimedia\WMPlayer`.
- `/UnRegExts` — удалить из реестра все настройки расширений проигрывателя. Восстановить их можно с помощью параметра `/RegExts`.

Shmgrate.exe

Программа используется при установке браузера Internet Explorer и других стандартных программ операционной системы Windows XP. Она имеет несколько параметров, с помощью которых можно отобразить или скрыть ярлык браузера на Рабочем столе или в меню Пуск, а также выполнить некоторые другие операции.

- `OCInstallHideIE` — скрыть с Рабочего стола и меню Пуск (из списка Программы) значки браузера Internet Explorer.
- `OCInstallReinstallIE` — переустановить значки браузера Internet Explorer на Рабочем столе и в меню Пуск. Аналогичных действий можно добиться и с помощью параметра `OCInstallUserConfigIE`.

- OCInstallShowIE — отобразить на Рабочем столе и в меню Пуск (в списке Программы) значки браузера Internet Explorer.
- Hide-WM — скрыть из меню Пуск (из списка Программы) значок программы Windows Messenger.
- Reinstall-WM — переустановить значок программы Windows Messenger в списке Программы меню Пуск.
- Show-WM — отобразить значок программы Windows Messenger в списке Программы меню Пуск.
- OCInstallHideVM, OCInstallShowVM, OCInstallReinstallVM — эти три параметра соответственно скрывают или отображают значок (первый и второй параметры), а также переустанавливают консоль Java (третий параметр).
- OCInstallCleanupInitiallyClear — очищает различные компоненты, указанные в ветви системного реестра HKEY_CURRENT_USER\Software\Microsoft\Active Setup\Installed Components\InitiallyClear. После очистки данная ветвь реестра удаляется.
- OCInstallUpdate — переписывает настройки почтового клиента Outlook Express и браузера Internet Explorer, определяющие версии этих программ и установленные для них компоненты. После выполнения данного параметра также создается ветвь реестра HKEY_CURRENT_USER\Software\Microsoft\Active Setup\Installed Components\InitiallyClear.
- OCInstallFixup — восстанавливаются настройки из ветви системного реестра HKEY_LOCAL_MACHINE\SOFTWARE\Clients\Mail\Outlook Express\InstallInfo.
- OCInstallUserConfigOE — отобразить значок программы Outlook Express в списке Программы меню Пуск. Аналогичное действие можно поризвести и с помощью параметров OCInstallShowOE или OCInstallReinstallOE.
- OCInstallHideOE — скрыть значок программы Outlook Express в списке Программы меню Пуск.
- AddConfigurePrograms — добавить в меню Пуск элемент Выбор программ по умолчанию.
- Fix-HTML-Help — создать папку %systemdrive%\Documents and Settings\All Users\Application Data\Microsoft\HTML Help.
- MoveAndAdjustIconMetrics — вызов программы с этим параметром устанавливает значения по умолчанию для следующих параметров строкового типа: IconSpacing, IconTitleWrap и IconVerticalspacing — из ветви реестра HKEY_CURRENT_USER\Control Panel\Desktop\WindowMetrics.

ПРИМЕЧАНИЕ

Этот параметр перед своим выполнением просматривает содержимое ветви реестра HKEY_CURRENT_USER\Control Panel\Desktop, и если найдет в ней параметры аналогичные устанавливаемы, то создает параметры на основе их содержимого.

- `Fix-Folders Fix-Curs` — восстановить по умолчанию значения параметров из ветви реестра `HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\User Shell Folders`.

Regsvr32.exe

Программа предназначена для регистрации библиотек и ActiveX-объектов. Но, кроме этого, она поддерживает множество других функций. Рассмотрим некоторые из них.

- `/i:«ôàéë ñöåíàðíîãî êîìïîíåíòà Windows (ðàñøèðåíèå .sct)» scrobj.dll` — зарегистрировать данный сценарный компонент в реестре.
- `/u /u /i:«ôàéë ñöåíàðíîãî êîìïîíåíòà Windows (ðàñøèðåíèå .sct)» scrobj.dll` — удалить информацию о данном сценарном компоненте из реестра.
- `/s /n /i:U shell32.dll` — добавить к панели быстрого запуска ярлык Свернуть все окна. Устанавливаются также по умолчанию пути к стандартным папкам Windows.
- `/s /u «áèáëèîòåêà»` — удалить из реестра информацию о данной библиотеке.
- `/u zipfldr.dll` — отключить отображение ZIP-архивов в виде обычных папок.

Другие команды

В Windows также используются следующие команды.

- `%WINDIR%\system32\OOBE\msoobe.exe /a` — вызвать окно активизации Windows.
- `Fsquirt.exe -receive` — получить файл от Bluetooth-устройства.
- `Fsquirt.exe -send` — отправить файл Bluetooth-устройству.
- `Wmplayer.exe /SHELLHLP_V9 Play` — отобразить окно Проигрывателя Windows Media , но скрыть окно листа проигрывания.
- `iexpress.exe /Q /N` — вызвать Мастер создания самораспаковывающихся пакетов с невозможностью выхода из него до того, как будет создан пакет.
- `shrpubw.exe /s «êîìïüþòåð»` — создать общедоступную папку на удаленном компьютере.

Приложение 3. Файлы справки Windows

Приложение содержит краткий список наиболее интересных справочных файлов, поставляемых вместе с Windows. Согласитесь, иногда бывают моменты, когда срочно нужно о чем-либо прочитать. Причем когда-то давно вы уже встречали информацию об этом в одном из окон справки, но в данный момент совершенно не знаете, где именно вы ее видели. Надеюсь, у вас не будет таких моментов, но если однажды надежды не сбудутся, то вам поможет нижеизложенная информация.

Центр справки и поддержки включает в себя большое количество начальной информации, но кроме того, может отображать страницы любых других справочных программ. Мы не будет говорить о том, как пользоваться поиском и другими его функциями — это довольно просто и эффективно, но иногда нужно найти информацию по теме, для которой поисковое слово сразу и не подберешь.

Например, как найти информацию об устранении неполадок?

Но перед тем, как углубиться в море ссылок, вспомним, как этими ссылками пользоваться, ведь если вы просто введете в диалоге Запуск программы команду `Helpctr`, то попадете на начальную страницу Центра справки и поддержки. Поэтому нужно воспользоваться параметром командной строки `–url`, который указывается после названия программы. После этого параметра вводится один из адресов, приведенных ниже. Это выглядит так: `helpctr –url àäðåñ`.

Неполадки

hcp://help/tshoot/tsdrive.htm

Здесь содержится мастер устранения неполадок с сетевыми адаптерами и дисками, с помощью которого можно попытаться решить вопросы неработоспособности привода компакт-дисков или DVD, дисковода или жесткого диска, сетевого адаптера или ленточного накопителя.

hcp://help/tshoot/tsgame.htm

Еще один мастер устранения неполадок. На этот раз он пытается устранить неполадки мультимедиа и игр, и если у вас действительно есть некоторые неполадки в этой области, то советую заглянуть в него, вдруг найдете то, что нужно.

hcp://help/tshoot/tsdisp.htm

Вызывает мастер устранения неполадок с монитором, с помощью которого можно попытаться решить следующие проблемы: мерцание или искажение экрана, установка монитора, невозможность установки разрешения больше 640 × 480, ошибки при видеозаписи или анимации.

hcp://help/tshoot/hdw_keyboard.htm

Вызвать мастер устранения неполадок клавиатуры.

hcp://help/tshoot/tsInputDev.htm

Вызвать мастер определения неполадок в работе модема, мыши, а также мастер устранения неполадок сканера, инфракрасного устройства, камеры.

Другие возможности Центра справки и поддержки

hcp://system/netdiag/dglogs.htm

Вызвать окно диагностики сети. Из него вы сможете настроить саму функцию диагностики, а также собрать информацию о сети, сетевых компонентах компьютера (модем, сетевая карта и т. д.), а также о версии Windows и конфигурации таких программ, как Outlook Express и Internet Explorer.

hcp://system/sysinfo/sysinfomain.htm

С помощью этой ссылки можно вызвать более общую страницу — она определяет раздел Сведения о компьютере. С его помощью можно просмотреть общую информацию о компьютере (количество памяти, объем жесткого диска, производительность процессора, название BIOS, адреса IP), оборудовании и программном обеспечении, установленном на компьютере (список устаревших приложений, процент использования логических дисков и состояние использования различного оборудования системы), а также сведения о конкретных продуктах Microsoft (название продукта, его серийный номер и программы, автоматически запускаемые во время входа пользователя в систему).

hcp://services/centers/support?topic=hcp://system/sysinfo/sysHealthInfo.htm

Если предыдущая страница слишком обширна, то с помощью этой ссылки можно открыть страницу отображения сведений о системе (вторая ссылка предыдущей страницы). Эта ссылка находится в каталоге `%systemroot%\PCHEALTH\HELPCTR\System\sysinfo`.

hcp://system/sysinfo/RSoP.htm

С помощью этой ссылки можно вызвать мастер вывода результирующей политики (аналог оснастки `rsop.msc`).

hcp://system/updatectr/updatecenter.htm

Вызвать страницу центра Windows Update, на которой вы сможете узнать о новинках в области заплат и дополнений к Windows. Следует также учитывать, что для ее работы необходим доступ к Интернету.

hcp://system/compatctr/compatmode.htm

Если вам необходимо настроить параметры совместимости старых программ с Windows XP, то этот мастер был создан специально для вас. Конечно, Microsoft не гарантирует, что этот мастер является панацеей от всех болезней такого рода. Но тем не менее попробовать стоит.

hcp://services/subsite?node=_System_/Tools_Center&topic=hcp://system/blurbs/tools.htm

Вызвать страницу отображения служебных программ, с помощью которой можно запустить такие программы, как Удаленный помощник, Сведения о системе, Очистка диска, Дефрагментация диска, Архивация, Настройка системы и т. д. Эта и другие подобные ссылки находятся в каталоге `%systemroot%\PCHEALTH\HELPCTR\System\blurbs`.

Другие CHM-файлы

Кратко познакомимся с оставшимися CHM-файлами каталога `%systemroot%\help`. Их также можно вызвать из программы Центр справки и поддержки с помощью функции поиска.

Стандартные программы

Следующие файлы справки относятся к стандартным программам.

- `charmap.chm` — посвящен программе Таблица символов (`charmap.exe`) и тому, зачем вообще нужна эта программа (с помощью этой программы вы сможете вставить необходимый вам символ любого шрифта Windows, MS-DOS или Unicode в текстовую программу, а также узнать номер необходимого вам символа Unicode).
- `eudcedit.chm` — кроме таблицы символов, Windows содержит еще и программу, позволяющую самому создавать символы (`eudcedit.exe`) (после создания символа его можно вставить с помощью описанной выше программы `charmap`, выбрав в списке Шрифт позицию Все шрифты (личные символы)). Именно о ней и рассказывается в данном файле справки.

ПРИМЕЧАНИЕ

Сведения о файлах, в которых сохраняются пользовательские символы, содержится в ветви реестра HKEY_CURRENT_USER\EUDC.

- `clipbrd.chm` — еще одной программой, имеющей свой файл справки, является буфер обмена Windows (`clipbrd.exe`), в котором вы сможете увидеть

текущее содержимое буфера или сохранить это содержимое в файле для будущего использования. Эта программа также позволяет просмотреть содержимое буфера удаленных компьютеров.

- `ddeshare.chm` — описывает работу программы «Общие ресурсы DDE», с помощью которой можно просмотреть открытые в данный момент ресурсы DDE локального или любого удаленного компьютера.
- `dialer.chm` — описывает работу с программой `dialer.exe`, в чем-то сходной с рассмотренной нами выше программой NetMeeting. Программа позволяет общаться с помощью микрофона или видеокамеры с другими участниками сети, домена или Интернета. Единственное, чего она не может, так это работать в качестве чата — набирать сообщения с клавиатуры в ней нельзя.
- `winchat.chm` — зато это можно сделать в отдельной программе — `winchat.exe`, как раз для этого и предназначенной. Программа работает в реальном времени и позволяет общаться только с одним пользователем, то есть ваш собеседник видит сам процесс вашего набора сообщения, а не уже готовое сообщение.
- `hypertrm.chm` — здесь есть возможность прочитать о приложении Hyper Terminal (`hypertrm.exe`), предназначенной для работы с подключениями telnet или досками BBS.
- `drwtsn32.chm` — здесь можно прочитать об отладчике ошибок приложений «Доктор Ватсон» (`drwtsn32.exe`).
- `dxdiag.chm` — можно прочитать сведения о работе со средством диагностики DirectX (`dxdiag.exe`).
- `msconfig.chm` — позволяет прочитать сведения о работе с программой настройки системы (`msconfig.exe`). Здесь вы узнаете о режимах запуска операционной системы, а также о вкладке Автозагрузка данной программы.
- `msinfo32.chm` — здесь можно прочитать сведения о работе с программой `msInfo32.exe`, отображающей данные о компьютере и оборудовании, установленном на нем.
- `ntbackup.chm` — содержит сведения о работе с программой `ntbackup.exe`, предназначенной для архивирования файлов.
- `mstsc.chm` — включает в себя сведения о работе с программой `mstsc.exe`, предназначенной для подключения к удаленному Рабочему столу.
- `magnify.chm` — содержит сведения о работе с экранной лупой — одним из средств комплекта специальных возможностей Windows. Другим средством является экранная клавиатура — работе с ней посвящен файл справки `osk.chm`. Можно также прочитать сведения о программе Экранный диктор — для этого предназначен файл `reader.chm`.
- И еще один файл справки, относящийся к настройке специальных возможностей — `utilmgr.chm`. Он описывает Диспетчер служебных программ, вызываемый нажатием комбинации клавиш Windows+U и содержащий сведения о текущей работе программ, относящихся к настройке специальных возможностей.

Панель управления

Следующие файлы справки относятся к панели управления.

- access.chm — рассказывает о настройке Windows для людей с различными нарушениями слуха, зрения или подвижности. Он посвящен апплету Специальные возможности, который можно вызвать командой access.cpl. В каталоге также присутствует файл справки accessib.chm, который содержит общие сведения, помогающие найти дополнительные программы для настройки специальных возможностей.
- addremov.chm — здесь содержатся сведения о работе с апплетом Установка и удаление программ (appwiz.cpl): как общие сведения, так и инструкции использования отдельных вкладок апплета.
- camera.chm — поможет вам справиться с вашей первой цифровой камерой или сканером. Из него вы узнаете о том, как подключить сканер или камеру к компьютеру, как получить от них изображения и как указать программу, которая будет обрабатывать эти изображения.
- fxsclnt.chm — если вместо сканера или камеры вы купили факс, то не расстраивайтесь, ведь данный файл содержит сведения по всем вопросам, которые могут у вас появиться при работе с новой покупкой: установка и настройка службы факсов, передача и прием факса, содержимое страниц факса и многое другое. К тому же после настройки факса вы можете заняться редактированием или рисованием титульной страницы — именно этому вопросу посвящен файл справки fxscover.chm.
- mode.chm — общие сведения о модемах, установке модема и настройке его конфигурации с помощью апплета Телефон и модем (telephon.cpl) — вот те сведения, которые вы сможете узнать, прочитав данный файл справки.
- datetime.chm — поможет справиться с апплетом Дата и время (timedate.cpl) и его функцией синхронизации времени с сервером Интернета или контроллером домена.
- display.chm — о работе с апплетом Свойства: Экран (desk.cpl) можно прочитать здесь.
- pwrmn.chm — в этом файле справки описывается работа с апплетом Электропитание (powercfg.cpl).
- folderop.chm — еще одним апплетом, работу с которым решила описать корпорация Microsoft, является апплет Свойства папки. Из этого файла справки вы сможете узнать о том, как работать с вкладкой Типы файлов этого апплета, а также как изменять общий вид оболочки Windows.
- fonts.chm — типы шрифтов, добавление и удаление шрифтов из системы, распечатка шрифтов — вот те темы, описание которых содержит данный файл справки.
- hardware.chm — типы оборудования, способы их установки, а также сведения о профилях оборудования — это те сведения, которые вы сможете получить из данного файла справки.

- input.chm — если вам нужно узнать о службах текстового ввода и их настройках с помощью апплета Язык и региональные стандарты (intl.cpl), о работе с планшетом для рукописного ввода, функциями голосового ввода и о многом другом, то этот файл справки для вас.
- joy.chm — раз уж мы затронули службы текстового ввода, то следует продолжить рассказом об апплете Игровые устройства (joy.cpl), с помощью которого можно настроить параметры подключения различных джойстиков.
- keyb.chm — описывает работу с апплетом Клавиатура.
- mouse.chm — справочный файл по настройке мыши с помощью апплета Мышь (main.cpl).
- sysdm.chm — данная справка посвящена работе с апплетом Система (sysdm.cpl).

Оснастки

Следующие файлы справки относятся к оснасткам.

- mmc.chm — для начала можно почитать общие сведения о консоли управления безопасностью mmc: это работа с консолью, добавление или удаление оснасток, а также описание терминов, необходимых для понимания работы консоли управления безопасностью.
- audit.chm — аудит событий безопасности с использованием групповой политики, а также вкладки Безопасность локальных файлов и папок — именно этой теме посвящен данный файл справки. Он содержит как общие сведения о том, для чего нужен аудит, и терминах, применяемых при его описании, так и инструкции по использованию оснастки групповой политики для установки аудита. Файл также хранит некоторые сведения о просмотре журнала безопасности.
- els.chm — но в предыдущем файле справки о журнале безопасности сказано мало. Зато этот файл справки полностью компенсирует получившуюся недосказанность — он содержит общие сведения об оснастке Просмотр событий (eventvwr.msc). Аналогичные сведения можно просмотреть в файле справки evconcepts.chm.
- certmgr.chm — Сертификаты, именно об этой оснастке и идет речь в данном файле справки. Он хранит как инструкции по работе с самой оснасткой, так и общие сведения о терминах, которые нужно знать для свободного общения с функциями импорта и экспорта сертификатов. Существует аналог данного файла справки — cmconcepts.chm.
- compmgmt.chm — в этом файле справки речь идет об оснастке Управление компьютером. Уже стало традицией, что он содержит как инструкции по работе с оснасткой, так и описание терминов, которые используются для этого.
- conf1.chm — описывает параметры групповой политики, относящиеся к настройкам ограничений для программы NetMeeting 3.01. Можно также прочитать об изменении интерфейса Internet Explorer с помощью групповых политик. Для этого предназначен файл справки ieakmmc.chm. Но если вам нужно

выполнить настройку ограничений для Internet Explorer, то предыдущий файл справки вам не поможет — вам нужен файл inetres.chm.

- wmplay.chm — файл справки о параметрах ограничений для Проигрывателя Windows Media .
- wuau.chm — файл справки, содержащий параметры ограничений групповой политики для автоматического обновления Windows.
- rrc.chm — описание регистрации событий завершения работы.
- gpedit.chm — но если для вас предыдущих файлов мало, то этот файл справки содержит общие сведения об оснастке Групповая политика (gpedit.msc). А вот сведения и описания параметров ограничений, настраиваемых групповой политикой, можно прочитать в файле справки system.chm.
- rsop.chm — здесь описывается работа с оснасткой Результирующая политика (rsop.msc).
- safer.chm — посвящен оснастке управления шаблонами безопасности. Его аналогами являются следующие файлы справки: saferconcepts.chm, sce.chm и sceconcepts.chm.
- lpe.chm — можно прочитать о локальной политике безопасности. Как раз для этого и предназначен данный файл справки. Аналогом этого файла является файл справки lpeconcepts.chm.
- defrag.chm — описывает работу с оснасткой dfrg.msc, предназначенной для выполнения дефрагментации отдельных локальных дисков компьютера или всех томов жесткого диска. Имеется также аналог данного файла справки — он называется dkconcepts.chm.
- devmgr.chm — описывает работу с оснасткой devmgmt.msc, предназначенной для просмотра сведений об установленных на компьютере устройствах, а также изменения используемых ими драйверов или удаления сведений об устройстве.
- diskmgmt.chm — предназначен для описания работы с оснасткой управления дисками (diskmgmt.msc). С ее помощью можно отформатировать логический диск, сделать его активным, изменить его букву, удалить сам логический диск или создать новый. Оснастка также позволяет просмотреть свойства диска или открыть содержимое диска в Проводнике.
- file_srv.chm — описывает работу с оснасткой fsmgmt.msc, предназначенной для создания общих папок. Но кроме этой оснастки, файл описывает создание общих папок с помощью специальной программы командной строки, а также с помощью диалогового окна свойств папки.
- is.chm — содержит море информации о службе индексирования и работе с ней с помощью соответствующей оснастки консоли mmc (ciadv.msc). Аналогом этого файла является файл isconcepts.chm.
- localsec.chm — и еще одно описание оснастки. На этот раз Локальные пользователи и группы (lusrmgr.msc).

- mail.chm — описывает оснастку администрирования почтовой службы SMTP, а также общие сведения об этой службе. Правда, почему-то для операционной системы Windows 2000.
- mpconcepts.chm — содержит сведения о работе с оснасткой Производительность (perfmon.msc).
- newfeat1.chm — включает в себя сведения об инструментарии управления WMI (WMIMGMT.MSC), предназначенном для сбора сведений о локальном или удаленном компьютере. Есть также возможность просмотреть сведения о программе wbemtest.exe, предназначенной для тестирования WMI. Они отображаются после выбора файла справки wbemtest.chm. Но и это еще не все — файл справки wmic.chm включает в себя сведения о работе с программой командной строки, предназначенной для работы с WMI.
- rsm.chm — хранит сведения о работе с оснасткой Съемные ЗУ (ntmsmgr.msc), предназначенной для администрирования съемных носителей.
- scm.chm — включает в себя сведения об оснастке Анализ системной безопасности. Его аналогом является файл scmconcepts.chm.
- secsetconcepts.chm — содержит сведения о работе с оснасткой Управление параметрами безопасности. Его аналогом является файл secsettings.chm.
- sys_srv.chm — включает в себя сведения об оснастке services.msc, отображающей службы, установленные на локальном компьютере. Из него вы не только узнаете о том, как запустить, остановить или просмотреть зависимости службы, но и получите сведения о настройке служб по умолчанию в операционной системе Windows.

Администрирование

Следующие справочные файлы относятся к администрированию.

- aclui.chm — содержит сведения о настройке доступа к файлам и папкам с помощью вкладки Безопасность. Он также описывает привилегии и права на вход в систему (и установленные по умолчанию привилегии), которые можно настроить с помощью оснастки шаблонов безопасности или групповой политики (gpedit.msc). При этом файл хранит как общие сведения о терминах, применяемых для описываемых операций, так и инструкции по установке разрешений в Windows XP.
- dskquoui.chm — кроме настройки ограничений на доступ к файлам и папкам, Windows предоставляет возможность установки дисковых квот для отдельных пользователей с помощью вкладки Квота диалога Свойства: Диск или специальной программы командной строки. Вот об этом и рассказывается в данном файле справки.
- comexp.chm — посвящен настройке и администрированию службы компонентов, программ DCOM и COM+. Из этого файла вы узнаете практически все о подобных программах, но, чтобы понять то, что в нем написано, нужно не понаслышке знать об этой службе и необходимости ее администрирования.

- `iis.chm` — можно прочитать о настройке, установке и администрировании служб IIS (информационные службы Интернета). Для этого и предназначен данный файл справки (или файл справки `iismmc.chm`, имеющий чуть больше сведений).
- `nwdoc.chm` — описывает службу клиентов NetWare: ее установки и настройки.
- `wshconcepts.chm` — содержит начальные сведения о сервере сценариев Windows — о работе с командной строкой, а также описание объектов, которые им поддерживаются.
- `msmq.chm` — включает в себя сведения о службе очереди сообщений (общие сведения, настройка, отличия между расположением программ настройки данной службы в операционных системах Windows NT и Windows XP), с помощью которой можно передавать короткие текстовые сообщения по локальной сети. Этот файл вам может быть полезен только в том случае, если вы используете Active Directory.
- `omc.chm` — если вы работаете в среде Active Directory, то вам может понадобиться не только предыдущая справка, но и текущая — она содержит сведения о методе поиска в Active Directory.
- `ODBCJET.HLP` — хранит справку о работе с базой ODBC и расположен в каталоге `%systemroot%\system32`. Следует только учитывать, что этот файл написан на английском языке.
- `atm.chm` — содержит сведения об установке ATM-адаптера и работе с ним: о настройке, установке службы, конфигурации подключения, а также о программах, предназначенных для администрирования ATM-подключений.
- `blutooth.chm` — именно этот файл справки и используется первой рассмотренной нами ссылкой — он включает в себя все те сведения, которые можно получить и в Центре справки и поддержки, но в этом файле они более упорядочены.
- `infrared.chm` — содержит сведения о беспроводных сетях, которые мы уже рассматривали, но, кроме этих сведений, он хранит и дополнительную информацию.
- `Ipv6.chm` — хранит основные сведения о протоколе IP шестой версии и поддержке его в Windows XP: общие отличия от четвертой версии протокола IP, а также способ установки данного протокола.
- `ipsecconcepts.chm` — здесь вы сможете узнать о протоколе IPSec, способном обеспечить безопасные соединения с другими компьютерами локальной сети или Интернета (аналогом этого файла является файл справки `ipsecsnp.chm`).
- `migwiz.htm` — краткие сведения о создании прямого подключения между двумя компьютерами.
- `telnet.chm` — содержит общие сведения о клиенте и сервере telnet, а также о синтаксисе программ, необходимых для работы с этим протоколом.
- `netcfg.chm` — хранит море сведений о работе с сетью в Windows XP, а также общие сведения о различных терминах, применяемых в данной области IT.
- `network.chm` — если предыдущий файл содержал общие сведения о работе с сетью, то этот файл справки посвящен вопросу создания сети в Windows XP.

Отдельно хотелось бы сказать о трех файлах справки, которые хранят все сведения, описанные нами выше, а также некоторые из тех сведений, которые будут рассмотрены в следующем разделе этого приложения. Это файлы справки `cpanel.chm`, `windows.chm` и `glossary.chm`. Первый из них содержит общие сведения о понятиях, упоминаемых в CPL-файлах панели управления, а также общую информацию об использовании этих файлов. Он также включает в себя дополнительные сведения о сетевых протоколах, рассмотренных нами ранее. Второй файл содержит все сведения о работе с Windows XP и программами командной строки. По своей функциональности он является аналогом Центра справки и поддержки, но если в данном центре упор сделан на начинающих пользователей и получить доступ к «продвинутой» информации можно только в контексте поиска, то файл `windows.chm`, кроме функции поиска, хранит полный указатель всех тем, описание которых в нем находится. Последний же файл просто содержит список терминов, описание которых можно прочитать благодаря заботе технического отдела Microsoft.

Мы уже говорили выше о таких обобщенных файлах справки, как `cpanel.chm` и `windows.chm`, но это еще не все файлы справок подобного рода — к ним также можно отнести файл `admtools.chm`. Он содержит указатель программ, предназначенных для администрирования компьютера, но, в отличие от предыдущих файлов справки, он не хранит никакой информации, кроме возможности открытия Центра справки и поддержки, других справочных файлов или самих оснасток и апплетов, посвященных необходимой теме.

Другие файлы

Напоследок еще несколько справочных файлов.

- `htmlref.chm` — вы решили начать изучение HTML-кодинга? Тогда этот файл справки для вас — он содержит общие сведения об элементах, поддерживаемых Internet Explorer, и их атрибутах.
- `hschelp.chm` — а еще, как это ни парадоксально звучит, можно прочитать справку о Центре справки и поддержки. Или справку о средствах справки — `nthelp.chm`.
- `spad.chm` — хранит справку о функции Выбор программ по умолчанию.
- `intellimirror.chm` — рассказ о функции IntellMirror (с ее помощью можно устанавливать или обновлять по сетевому соединению программы или саму операционную систему Windows локального компьютера).
- `keyshort.chm` — очень интересный файл справки — он описывает все сочетания клавиш быстрого доступа, используемые в операционной системе Windows XP.
- `ntcmds.chm` — также очень интересный файл справки. Он содержит сведения о программах командной строки Windows XP: названия, для чего они предназначены, а также параметры их запуска. Файл также хранит список программ командной строки, которые уже не поддерживаются в операционной системе.
- `Default.htm` — находится в каталоге `%systemroot%\HELP\Tours\htmlTour` и является начальной страницей приветствия Windows. Можно просмотреть

еще одно приветствие — более красивое, но уже на английском языке. Оно расположено в файле `tour.exe` каталога `%systemroot%\HELP\Tours\mmTour`.

Конечно, это не все справочные файлы, которые могут находиться на вашем компьютере, ведь, кроме стандартных файлов, сегодня практически каждая программа предлагает свои. Именно поэтому было решено привести несколько ветвей реестра, содержащих сведения о зарегистрированных в системе справочных файлах.

- `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\Help` — содержит сведения о названии HLP-файла (названия строковых параметров, расположенных в этой ветви) и пути, по которому он находится (значения этих параметров).
- `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\HTML Help` — если предыдущая ветвь реестра содержала сведения о справочных файлах с расширением HLP, то эта ветвь реестра хранит сведения о зарегистрированных справочных файлах с расширением CHM. Формат хранения сведений такой же: название параметра — это название файла, а значением параметра является каталог, в котором файл хранится.

Приложение 4. ActiveX-объекты

Приложение содержит краткий список ActiveX-объектов, идентифицирующих различные команды или значки оболочки Windows. Кроме описания ActiveX-объекта, данное приложение определяет действия, которые можно выполнить с объектом в Windows: можно ли создавать папки на основе объекта, отобразить значок, вызывать с помощью команды Выполнить.

ПРИМЕЧАНИЕ

Если поле Диалог Запуск программы равно Да, то соответствующий ActiveX-объект можно вызвать с помощью команды диалога Выполнить ::{CLSID-номер объекта}. Если поле Создание папки равно Да, то копию соответствующего ActiveX-объекта можно создать следующим способом: создать папку и после ее названия ввести .{CLSID-номер объекта}.

- `{0DF44EAA-FF21-4412-828E-260A8728E7F1}`

  Объект: значок Панель задач и меню "Пуск".

  Диалог Запуск программы: нет.

  Создание папки: да.

- `{208D2C60-3AEA-1069-A2D7-08002B30309D}`

  Объект: значок Сетевое окружение.

  Диалог Запуск программы: да.

  Создание папки: да.

- `{20D04FE0-3AEA-1069-A2D8-08002B30309D}`

  Объект: значок Мой компьютер.

  Диалог Запуск программы: да.

  Создание папки: нет.

- `{21EC2020-3AEA-1069-A2DD-08002B30309D}`

  Объект: значок Панель управления.

  Диалог Запуск программы: нет, однако можно запустить, используя команду `::{20D04FE0-3AEA-1069-A2D8-08002B30309D}\::{21EC2020-3AEA-1069-A2DD-08002B30309D}`.

  Создание папки: да.

- {2227A280-3AEA-1069-A2DE-08002B30309D}

  Объект: значок Принтеры.

  Диалог Запуск программы: да.

  Создание папки: да.

- {2559a1f0-21d7-11d4-bdaf-00c04f60b9f0}

  Объект: значок Поиск.

  Диалог Запуск программы: нет.

  Создание папки: нет.

- {2559a1f1-21d7-11d4-bdaf-00c04f60b9f0}

  Объект: значок Справка и поддержка.

  Диалог Запуск программы: нет.

  Создание папки: да.

- {2559a1f2-21d7-11d4-bdaf-00c04f60b9f0}

  Объект: значок Безопасность Windows.

  Диалог Запуск программы: нет.

  Создание папки: нет.

- {2559a1f3-21d7-11d4-bdaf-00c04f60b9f0}

  Объект: значок Запуск программы.

  Диалог Запуск программы: нет.

  Создание папки: нет.

- {2559a1f4-21d7-11d4-bdaf-00c04f60b9f0}

  Объект: значок Интернет.

  Диалог Запуск программы: нет.

  Создание папки: да.

- {2559a1f5-21d7-11d4-bdaf-00c04f60b9f0}

  Объект: значок Outlook Express.

  Диалог Запуск программы: нет.

  Создание папки: да.

- {2559a1f7-21d7-11d4-bdaf-00c04f60b9f0}

  Объект: значок Установка и удаление программ, открываемый в разделе Выбор программ по умолчанию.

  Диалог Запуск программы: нет.

  Создание папки: нет.

- {2728520d-1ec8-4c68-a551-316b684c4ea7}

  Объект: значок Мастер настройки сети.

Диалог Запуск программы: нет.

Создание папки: да.

- {3c5c43a3-9ce9-4a9b-9699-2ac0cf6cc4bf}

  Объект: значок Мастер беспроводной сети.

  Диалог Запуск программы: нет.

  Создание папки: да.

- {450D8FBA-AD25-11D0-98A8-0800361B1103}

  Объект: значок Мои документы.

  Диалог Запуск программы: да.

  Создание папки: да.

- {645FF040-5081-101B-9F08-00AA002F954E}

  Объект: значок Корзина.

  Диалог Запуск программы: да.

  Создание папки: да.

- {6DFD7C5C-2451-11d3-A299-00C04F8EF6AF}

  Объект: значок Свойства папки.

  Диалог Запуск программы: да.

  Создание папки: да.

- {7007ACC7-3202-11D1-AAD2-00805FC1270E}

  Объект: значок Сетевые подключения.

  Диалог Запуск программы: да.

  Создание папки: да.

- {7A9D77BD-5403-11d2-8785-2E0420524153}

  Объект: значок Учетные записи пользователей.

  Диалог Запуск программы: нет.

  Создание папки: да.

- {7be9d83c-a729-4d97-b5a7-1b7313c39e0a}

  Объект: значок Программы, хранящий содержимое данной папки из профиля пользователя (%userprofile%\Ãëàâíîå ìåíþ) и содержимое папки для всех профилей (%systemdrive%:\Documents and Settings\All Users\Ãëàâíîå ìåíþ).

  Диалог Запуск программы: нет.

  Создание папки: да.

- {85BBD920-42A0-1069-A2E4-08002B30309D}

  Объект: значок Портфель.

Диалог Запуск программы: нет.

Создание папки: да.

- {871C5380-42A0-1069-A2EA-08002B30309D}

  Объект: значок Internet Explorer.

  Диалог Запуск программы: да.

  Создание папки: да.

- {992CFFA0-F557-101A-88EC-00DD010CCC48}

  Объект: значок Сетевые подключения.

  Диалог Запуск программы: нет.

  Создание папки: да.

- {AFDB1F70-2A4C-11d2-9039-00C04F8EEB3E}

  Объект: значок Папка автономных файлов.

  Диалог Запуск программы: нет.

  Создание папки: да.

- {BDEADF00-C265-11d0-BCED-00A0C90AB50F}

  Объект: значок Веб-папки.

  Диалог Запуск программы: нет.

  Создание папки: да.

- {D20EA4E1-3957-11d2-A40B-0C5020524152}

  Объект: значок Шрифты.

  Диалог Запуск программы: нет, однако можно вызвать с помощью команды ::{20D04FE0-3AEA-1069-A2D8-08002B30309D}\::{21EC2020-3AEA-1069-A2DD-08002B30309D}\::{D20EA4E1-3957-11d2-A40B-0C5020524152}.

  Создание папки: нет.

- {D20EA4E1-3957-11d2-A40B-0C5020524153}

  Объект: значок Администрирование.

  Диалог Запуск программы: нет, однако можно вызвать с помощью команды ::{20D04FE0-3AEA-1069-A2D8-08002B30309D}\::{21EC2020-3AEA-1069-A2DD-08002B30309D}\::{D20EA4E1-3957-11d2-A40B-0C5020524153}.

  Создание папки: нет.

- {D4480A50-BA28-11d1-8E75-00C04FA31A86}

  Объект: значок Новое место в сетевом окружении.

  Диалог Запуск программы: нет.

  Создание папки: да.

- {D6277990-4C6A-11CF-8D87-00AA0060F5BF}

  Объект: значок Назначенные задания.

  Диалог Запуск программы: да.

  Создание папки: да.

- {E211B736-43FD-11D1-9EFB-0000F8757FCD}

  Объект: значок Сканеры и камеры.

  Диалог Запуск программы: нет.

  Создание папки: нет.

- {FB0C9C8A-6C50-11D1-9F1D-0000F8757FCD}

  Объект: значок Сканеры и камеры.

  Диалог Запуск программы: нет.

  Создание папки: нет.

Вот, собственно, и все ActiveX-объекты Windows, которые открывают свои окна или диалоги. Но напоследок хотелось бы напомнить еще о нескольких командах, которые можно ввести в диалоге Запуск программы:

- shell:Desktop — открыть папку Рабочий стол текущего пользователя;
- shell:ControlPanelFolder — открыть папку Панель управления;
- shell:DriveFolder — открыть папку Мой компьютер.

Теперь перечислим некоторые из наиболее интересных ActiveX-объектов, определяющих дополнительные вкладки диалога Свойства или команды контекстного меню для файлов различных расширений.

- {7BA4C740-9E81-11CF-99D3-00AA004AE837}

  Объект: команда контекстного меню Отправить.

  Расширение: стандартный идентификатор AllFilesystemObjects.

- {645FF040-5081-101B-9F08-00AA002F954E}

  Объект: кроме значка Корзина, добавляет команду контекстного меню Очистить корзину.

  Расширение: HKEY_CLASSES_ROOT\CLSID\{645FF040-5081-101B-9F08-00AA002F954E}\shellex\ContextMenuHandlers.

- {f81e9010-6ea4-11ce-a7ff-00aa003ca9f6}

  Объект: вкладка Доступ диалога Свойства.

  Расширение: стандартный идентификатор Directory.

- {ef43ecfe-2ab9-4632-bf21-58909dd177f0}

  Объект: вкладка Настройка диалога Свойства.

  Расширение: стандартный идентификатор Directory.

- {7988B573-EC89-11cf-9C00-00AA00A14F56}

  Объект: вкладка Квота диалога Свойства.

  Расширение: стандартный идентификатор Drive.
- {513D916F-2A8E-4F51-AEAB-0CBC76FB1AF8}

  Объект: вкладка Совместимость диалога Свойства.

  Расширение: идентификатор исполняемых файлов.
- {4a7ded0a-ad25-11d0-98a8-0800361b1103}

  Объект: вкладки Папка назначения, Доступ и Общие диалога Свойства.

  Расширение: объект Мои документы.
- {1F2E5C40-9550-11CE-99D2-00AA006E086C}

  Объект: вкладка Безопасность диалога Свойства.

  Расширение: стандартный идентификатор Drive.
- {f81e9010-6ea4-11ce-a7ff-00aa003ca9f6}

  Объект: команда контекстного меню Общий доступ и безопасность.

  Расширение: стандартный идентификатор Folder.
- {F1B9284F-E9DC-4e68-9D7E-42362A59F0FD}

  Объект: команды контекстного меню Поставить в очередь и Добавить в список, добавляющие содержимое папок в текущий список воспроизведения Проигрывателя Windows Media.

  Расширение: нет, но для удобства можно создать в Folder.
- {D969A300-E7FF-11d0-A93B-00A0C90F2719}

  Объект: команды контекстного меню Создать и Упорядочить значки.

  Расширение: Directory\Background\shellex\ContextMenuHandlers.
- {CE3FB1D1-02AE-4a5f-A6E9-D9F1B4073E6C}

  Объект: команда контекстного меню Воспроизвести с помощью проигрывателя.

  Расширение: нет, но для удобства можно создать в Folder.
- {C2FBB631-2971-11d1-A18C-00C04FD75D13}

  Объект: команда контекстного меню Переместить в.

  Расширение: нет, но для удобства можно создать в Folder.
- {C2FBB630-2971-11D1-A18C-00C04FD75D13}

  Объект: команда контекстного меню Копировать в.

  Расширение: нет, но для удобства можно создать в Folder.
- {b8cdcb65-b1bf-4b42-9428-1dfdb7ee92af}

  Объект: команда контекстного меню Извлечь все.

  Расширение: стандартный идентификатор CompressedFolder.

- `{8DD448E6-C188-4aed-AF92-44956194EB1F}`

  Объект: команда контекстного меню Копировать на компакт-диск или устройство.

  Расширение: нет, но для удобства можно создать в `Folder`.

- `{09799AFB-AD67-11d1-ABCD-00C04FC30936}`

  Объект: команда контекстного меню Открыть с помощью.

  Расширение: идентификатор исполняемых файлов.

Приложение 5. Идентификаторы библиотеки shell32.dll

Приложение содержит перечень первых наиболее часто используемых идентификаторов, применяемых для отображения стандартных значков Windows (табл. П5.1).

Таблица П5.1. Содержимое библиотеки shell32.dll

Изображение значка	Индекс значка	Идентификатор значка	Описание
	0	–1	Определяет значок для незарегистрированных типов файлов, а также для тех файлов, которые не имеют своего значка (параметр (По умолчанию) раздела идентификатора DefaultIcon пуст)
	2	–3	Значок используется приложениями MS-DOS (идентификатор comfile) или программами Win32, не имеющими своего значка
	3	–4	Значок используется для отображения папок Windows и описывается стандартным идентификатором Directory, предназначенным для идентификации папок с файлами
	4	–5	Значок используется для отображения открытой в данный момент папки (отображается в строке заголовка окна)
	6	–7	Именно этот значок можно изменить на значок пятидюймовой дискеты
	7	–8	Значок используется для отображения съемных носителей. Например, к таким носителям можно отнести внешние жесткие диски или USB-носители
	8	–9	Значок используется для отображения дисков Windows и описывается стандартным идентификатором Drive

Изображение значка	Индекс значка	Идентификатор значка	Описание
	9	–10	Значок используется в системе для отображения дисков, ассоциированных с сетевым устройством (папкой или, как это ни странно, диском)
	10	–11	Значок говорит о том, что в данный момент соединение с сетевым устройством разорвано
	11	–12	Значок используется для отображения привода компакт-дисков системы в папке Мой компьютер, а также в адресной строке при открытии или сохранении файла. Если же вы применяете привод DVD, то будет использоваться значок с индексом 177 (он отличается от этого надписью DVD)
	15	–16	Ну, этот значок знает каждый. Именно его чаще всего можно встретить на Рабочем столе пользователя
	16	–17	Значок используется в качестве ярлыка установленного на вашем компьютере принтера
	17	–18	Значок используется при отображении ActiveX-объекта Сетевое окружение (CLSID-номер {208D2C60-3AEA-1069-A2D7-08002B30309D})
	19	–20	Значок используется для отображения папки Программы, (а также других стандартных папок Windows, вложенных в нее), расположенной в каталоге Главное меню профиля пользователя
	23	–24	Значок используется диалоговым окном Справка и поддержка (CLSID-номер {2559a1f1-21d7-11d4-bdaf-00c04f60b9f0}) для отображения в различных меню или как отдельный значок. Он же применяется HLP-файлами. Но при его переопределении HLP-файлы ведут себя довольно странно — иногда используют стандартный значок, иногда тот, который вы указали, а иногда даже бывает, что в одной папке HLP-файлы используют стандартный значок, а в другой — ваш

Продолжение ⇗

Таблица П5.1. Содержимое библиотеки shell32.dll (продолжение)

Изображение значка	Индекс значка	Идентификатор значка	Описание
	28	–29	Значок отображается в левом нижнем углу папки и используется для идентификации папки в качестве общедоступной. Его можно увидеть, только если вы принадлежите к группе Администраторы и если включена служба Сервер
	29	–30	Значок отображается в левом нижнем углу значка файла и указывает на что, что данный файл является лишь ярлыком файла, который физически расположен в другом месте
	30	–31	Значок также отображается в левом нижнем углу значка файла
	31	–32	Значок по умолчанию используется для отображения пустой Корзины (CLSID-номер ActiveX-объекта {645FF040-5081-101B-9F08-00AA002F954E})
	32	–33	Значок по умолчанию используется для отображения полной Корзины (CLSID-номер ActiveX-объекта {645FF040-5081-101B-9F08-00AA002F954E}). Для этих целей также применяется ActiveX-объект {5ef4af3a-f726-11d0-b8a2-00c04fc309a4} (Recycle Bin Cleaner)
	38	–39	Значок используется для отображения папки Шрифты в Панели управления
Прозрачный фон	От 49 до 52	От –50 до –53	Идентификаторы определяют полностью прозрачный значок. Например, если вы установите один из этих идентификаторов для значка Корзины, а потом скроете ее название с помощью параметра реестра, приведенного в гл. 8, то вы не сможете увидеть на Рабочем столе своей Корзины, хотя при наведении на нее указателя он будет принимать форму руки. Точно то же можно проделать и со значком Мой компьютер
	55	–134	Значок используется ActiveX-объектом {e17d4fc0-5564-11d1-83f2-00a0c90dc849}, принадлежащим диалогу Поиск (именно этот объект отображается в строке Адрес после нажатия комбинации клавиш Windows+F для вызова диалога Поиск)

Изображение значка	Индекс значка	Идентификатор значка	Описание
	69	–151	Значок используется для отображения в информационных файлах с расширением INF. Он описывается идентификатором inffile (сведения для установки). Он же применяется для отображения файлов конфигурации (идентификатор inifile)
	70	–152	Значок используется для отображения текстовых файлов (идентификатор txtfile).Он же применяется файлами макросов удаленного доступа к сети (идентификатор scpfile)
	71	–153	Используется для отображения пакетных файлов MS-DOS и определяется в идентификаторе batfile. Он же применяется командными сценариями Windows NT (идентификатор cmdfile)
	72	–154	Используется в Windows для отображения системных библиотек (DLL-файлов, идентификатор dllfile, и идентификатором system), а также может применяться такими идентификаторами, как appfix (заплаты), chkfile (восстановленные фрагменты файлов), cplfile (компонент панели управления, или апплет), dbfile (файл базы данных), drvfile (драйвер устройств) и др.
	73	–155	Значок используется для отображения файлов шрифтов (идентификатор fonfile)
	74	–156	Значок используется для отображения файлов шрифтов TrueType и применяет для своего определения идентификатор ttffile
	75	–157	Значок используется для отображения файлов шрифтов Type 1 и использует для своего определения идентификатор pfmfile
	85	–172	Значок определяет общую сетевую папку
	86	–173	Значок используется ActiveX-объектом {1A9BA3A0-143A-11CF-8350-444553540000} (Shell Favorite Folder) для своего отображения. Идентификатор –173 также применяется ActiveX-объектом {B005E690-678D-11d1-B758-00A0C90564FE} (DocFind Command)

Как описано выше, значок для папки Мой компьютер можно установить прозрачным, затем скрыть ее название. В результате вы не сможете увидеть Мой компьютер на Рабочем столе, хотя указатель при наведении на него будет принимать форму руки (рис. П5.1.).

Рис. П5.1. Результат скрытия значка Мой компьютер с помощью изображения с индексом 49 библиотеки shell32.dll

Таким образом, благодаря разнообразию идентификаторов, применяемых для отображения стандартных значков Windows, можно легко понять, какой именно файл (папка) находится перед вами.

Приложение 6. Содержимое компакт-диска

В этом приложении будет рассмотрено назначение программ, которые вы сможете найти на диске, прилагаемом к книге. Диск содержит набор программ для оптимизации работы компьютера и не только. На диске также находится база данных, разработанная автором книги, и листинги, приведенные в книге.

Программы

Итак, рассмотрим программы, расположенные на диске, и их назначение.

AnyDVD

Размер: 1,2 Мбайт.

Статус: коммерческая.

Страница программы: http://www.slysoft.com/en/anydvd.html.

Комментарий: просто отличная программа.

Помните, при рассмотрении оснастки Диспетчер устройств упоминалось о специальной функции контроля воспроизведения DVD, имеющих территориальные ограничения на воспроизведение? Говорилось, что вы можете изменить территориальную привязку своего DVD-привода только пять раз, после чего за вашим компьютером закрепится последняя привязка и вы не сможете ее изменить (так же, как не сможете просмотреть DVD, которые не предназначены для вашей привязки), даже если переустановите операционную систему.

Довольно решительный шаг разработчиков в борьбе с пиратством, согласитесь. Но, как бывает в большинстве случаев, защита взламывается через несколько месяцев после своего появления. Так же произошло и на этот раз. Программа AnyDVD предназначена для обхода следующих защит DVD: региональная защита диска как на уровне приложения, так и на аппаратном уровне (то, что было найдено в Диспетчере устройств), защита от копирования, основанная на нечитаемых секторах, аналоговая система защиты. Не знаю, действительно ли данная программа обходит региональную защиту дисков — в Украине такие диски встретишь нечасто. Тем не менее данная программа успешно показала себя на защите типа CSS, применяемой в Европе практически к каждому новому диску. После копирования дисков с защитой CSS без использования данной программы качество получаемых копий просто ужасающее. Если же копировать такие диски с помощью данной программы, то получается копия фильма, полностью аналогичная оригиналу.

ПРИМЕЧАНИЕ

Программа также может обходить CD Digital аудиозащиту дисков.

Но как же работает эта программа? Процесс ее установки длится несколько секунд и требует административных прав. После установки программы необходимо перезагрузить компьютер, чтобы она смогла работать. По этим признакам можно понять, что программа AnyDVD устанавливает свой низкоуровневый драйвер, который является посредником между операционной системой и непосредственно драйверами DVD-привода. Это очень большой плюс, повышающий удобство использования программы. Работать с ней очень просто. По умолчанию AnyDVD автоматически запускается при включении компьютера, о чем символизирует ее значок в области уведомлений (симпатичная мордочка лисички). Если вас это не устраивает, то можно отключить автозапуск программы при помощи контекстного меню значка в области уведомлений (команда Автостарт), хотя это не обязательно, так как программа практически не отбирает никаких системных ресурсов. С помощью контекстного меню значка в области уведомлений можно также вызвать диалог настройки программы (команда Настройки). С его помощью вы можете указать региональный код для диска, который хотите просмотреть (как правило, это не обязательно, так как AnyDVD автоматически определяет необходимый диску код региона при установке его в привод), а также указать те из видов защиты дисков, которые программа должна взламывать. Еще одной интересной командой контекстного меню значка в области уведомлений является команда Показать информацию. Если в данный момент в приводе установлен DVD, то после выбора этой команды перед вами отобразится диалог с информацией об обнаруженных на диске системах защиты (если система защиты обнаружена, то программа может ее взломать).

В принципе, вот и все особенности настройки программы. Больше вам ничего делать не нужно. Если программа в данный момент запущена, то вы можете смело просматривать или копировать на жесткий диск содержимое защищенных DVD обычными методами Windows — AnyDVD работает в автоматическом режиме и никаких дополнительных действий от пользователя не требуется.

Подводя итоги, можно сказать, что AnyDVD является очень удобным и успешным экземпляром программ такого рода. Огорчает в ней лишь одно — срок работы демо-версии, равный 21 дням, после чего программа прекращает работать.

CloneDVD

Размер: 5,11 Мбайт.

Статус: коммерческая.

Страница программы: http://www.slysoft.com/.

Комментарий: хорошая программа для копирования всего содержимого DVD или только его части.

Еще одна программа, предназначенная для копирования DVD от разработчиков SlySoft. Если AnyDVD работала в автоматическом режиме, то для копирования дисков на компьютер с помощью этой программы необходимо непосредственно запускать программу CloneDVD и, пройдя через несколько окон настройки параметров копирования диска, начинать сам процесс копирования.

Aston

Размер: 2,5 Мбайт.

Статус: коммерческая.

Страница программы: http://www.astonshell.com/.

Комментарий: отличная программа, хотя и с небольшими недостатками.

В конце книги вкратце был рассмотрен вопрос изменения содержимого файлов библиотек оболочки Windows для личного пользования — изменение названия меню Пуск и т. д. Если самостоятельно выполнять кардинальную настройку оболочки Windows с помощью редактирования DLL-файлов не хочется, то можно воспользоваться одной из уже готовых альтернативных оболочек для Windows XP. Именно примером таких программ и является Aston. После ее установки в меню Пуск появится пункт Aston, содержащий ссылки на два файла — Aston Master и Aston Swapper. С помощью Aston Master выполняется настройка оболочки Aston, а с помощью Aston Swapper выполняется переход между стандартной оболочкой Windows и оболочкой Aston. После запуска Aston Swapper необходимо выбрать оболочку, которая будет использоваться, после чего нужно указать учетную запись пользователя, для которого эта оболочка будет применяться. Затем отобразится диалог окончания смены оболочки. Если в этом диалоге не снять флажок Logoff, то после закрытия данного диалогового окна будет выполнен автоматический выход из системы и вход в нее под новой оболочкой.

Теперь поговорим о настройке оболочки Aston, то есть о файле Aston Master. С его помощью можно очень точно настроить Рабочий стол пользователя: цвета, размер и название применяемого шрифта (на взгляд автора, по умолчанию в оболочке Aston применяется очень мелкий и неудобный шрифт), прозрачность панелей. Отдельно стоит сказать о панелях — их четыре. Во-первых, стандартная панель инструментов Windows в нижней части экрана (которую можно сделать полупрозрачной). Во-вторых и в-третьих, панели с левой и с правой стороны экрана. Они реализованы в виде набора кнопок (можно сделать полупрозрачными), автоматически скрываемых с экрана. Стоит сказать и о своеобразной записной книжке, доступ к которой можно получить с левой панели, — очень удобное решение, хотя и напоминает Блокнот. И в-четвертых, набор подпапок в самом верху экрана. По умолчанию верхняя панель разбита на отдельные подпапки (графика, мультимедиа и т. д.), содержащие списки наиболее часто используемых программ (даже если у вас нет соответствующих программ). Вы сами можете добавить как свои подпапки, так и отредактировать элементы уже существующих подпапок.

Собственно, это все — кроме настроек Рабочего стола, ничего не изменяется. Тот же Проводник и те же функции. Другими словами, Aston нельзя назвать полноценной оболочкой, просто небольшой надстройкой для explorer.exe, изменяющей Рабочий стол оболочки. И в этом амплуа Aston работает отлично. Рабочий стол хорошо продуман и даже по умолчанию очень красиво исполнен (хотя шрифт стоит изменить). Но есть у Aston и небольшой недостаток. В начале работы с программой была указана оболочка, применяемая для данной учетной записи (программа Aston Swapper). Иными словами, можно подумать, что если вы настроите оболочку Aston, то это никоим образом не повлияет на стандартную оболочку Windows, к которой можно будет перейти с помощью той же программы Aston Swapper. Но на самом деле все не совсем так. Например, стандартная оболочка Windows будет использовать те же цвета, которые применяются в Aston, а также те же наборы указателей. Поэтому перед использованием и настройкой Aston желательно выполнить резервирование всего реестра или отдельно ветви HKEY_CURRENT_USER — если изменения в стандартной оболочке после применения Aston вам не понравятся, то всегда можно будет оперативно восстановить те настройки, которые использовались вами ранее.

Подведем итог. Довольно хорошая программа, настроив которую соответствующим образом и привыкнув к ней, вы, скорее всего, будете пользоваться только ею.

FlyakiteOSX 3

Размер: 30,2 Мбайт.

Статус: бесплатная.

Страница программы: http://osx.portraitofakite.com/boot.htm.

Если вы являетесь поклонником операционной системы MacOs, но по странным обстоятельствам всегда пользовались Windows, то эта программа специально для вас. Она преобразует оболочку Windows в оболочку MacOs, изменяя DLL-библиотеки системы.

Jv16 PowerTools 2005

Размер: 2,11 Мбайт.

Статус: коммерческая.

Страница программы: http://www.macecraft.com/jv16powertools2005/.

Если использовать альтернативную оболочку Windows не хочется (пусть и ограничивающуюся лишь новым Рабочим столом), то можно более тонко настроить родную оболочку Windows. Для этого пригодится программа jv16 PowerTools 2005. Она предназначена для оптимизации работы компьютера и может следующее.

- Удалять программы или отдельные компоненты программы (раздел Software Manager и вкладка Add/Remove program раздела Registry Manager).
- Удалять зарегистрированные в системе типы файлов, удалять содержимое списка меню Создать контекстного меню Рабочего стола или Проводника Windows,

удалять команды контекстного меню файлов различных расширений, удалять содержимое контекстного меню Internet Explorer, удалять содержимое диалога Открыть с помощью, удалять содержимое списка Найти меню Пуск. Все эти действия выполняются с помощью раздела Registry Manager.

- Чистить реестр.
- Выполнять поиск в реестре или создавать его снимки (например, можно сделать снимок реестра до установки программы и сравнить его с реестром после ее установки, чтобы посмотреть на изменения, внесенные программой).
- Искать файлы в файловой системе компьютера, искать дубликаты файлов (не всегда работает правильно) или выполнять очистку файловой системы от файлов с расширениями TMP, TEMP, GID, CHK, ~* и т. д.
- Анализировать содержимое меню Пуск на предмет неработающих ярлыков и многое другое.

Стоит отдельно сказать о том, что большинство операций удаления могут сопровождаться архивированием, то есть вы в любой момент сможете восстановить удаленные программой элементы.

Подведем итог. Интересная программа, хотя ей не хватает функциональности. Например, кроме удаления содержимого списка Создать контекстного меню Рабочего стола, можно было бы реализовать и способы добавления в этот список своих команд.

IconPackager

Размер: 11,5 Мбайт.

Статус: коммерческая.

Страница программы: http://www.iconzone.com/.

Можно также воспользоваться этой программой. Она предназначена для изменения стандартных значков оболочки Windows на свои собственные. Но, кроме этого, программа позволяет настроить некоторые параметры реестра, относящиеся к работе со значками. Еще одной интересной возможностью программы является перезапись кэша значков. Помните, когда были рассмотрены возможности изменения стандартных значков Windows с помощью реестра, упоминалось о том, что изменения значков могут произойти не сразу, а через некоторое время, когда система перепишет кэш значков. Данная программа может самостоятельно переписывать кэш значков.

RightClick

Размер: 2,44 Мбайт.

Статус: коммерческая.

Страница программы: http://www.stardock.com/products/rightclick/.

С помощью этой программы можно заменить стандартное контекстное меню Рабочего стола. При этом она позволяет изменить не только цветовое оформление контекстного меню, но и его полупрозрачность и команды, отображаемые в нем. По умолчанию к контекстному меню добавляются команды Run, Find, Program. Думаю, объяснять назначение этих команд не надо — все они являются стандартными командами меню Пуск.

Плюсом программы является практически мгновенное отображение контекстного меню, тогда как отображение стандартного контекстного меню Windows может занять несколько секунд.

Rainlendar

Размер: 950 Кбайт.

Статус: бесплатная.

Страница программы: http://vapaa.dc.inet.fi/~rainy/index.php?pn=projects&project=rainlendar.

Программа является красивым календарем, отображаемым на Рабочем столе. Этот календарь поддерживает так модную нынче полупрозрачность, а также многие другие функции настройки.

RegSnap

Размер: 1,63 Мбайт.

Статус: коммерческая.

Страница программы: http://www.lastbit.com/.

Если рассмотренная ранее программа jv16 PowerTools умела делать довольно много вещей, среди которых была возможность создания снимка реестра, то данная программа предназначена только для одного — именно для создания снимка реестра или всей системы в целом. Но справляется она с этим очень хорошо. Если jv16 PowerTools для создания снимка корневых разделов HKEY_CURRENT_USER и HKEY_LOCAL_MACHINE требовалось несколько минут, то программе RegSnap для этого требуется примерно 5 секунд. При этом содержимое снимка можно просмотреть (ветви, параметры и значения параметром реестра), тогда как в jv16 PowerTools снимок можно сравнить только с другим снимком.

RegWorks

Размер: 980 Кбайт.

Статус: бесплатная.

Страница программы: http://www.regwrks.com.

Раз уж зашла речь о реестре, тем более что большая часть книги посвящена именно ему, то нужно описать и эту программу. Она является альтернативой стандартному редактору реестра Windows `regedit.exe`. RegWorks 1.3.3 умеет делать все, что и стандартный редактор реестра, но, кроме этого, умеет следующее: сравнение ветвей реестра, мониторинг изменения системы, запись всех изменений в архив (для возможности восстановления изменений, если что-то пойдет не так). Этот редактор реестра также включает в себя справочник по параметрам реестра, выполненный отдельно от самого редактора.

ExamDiff Pro

Размер: 1,73 Мбайт.

Статус: коммерческая.

Страница программы: http://www.prestosoft.com/.

Иногда может понадобиться сравнить изменения в коде двух файлов после выполнения какого-либо действия. Именно этим и занимается программа ExamDiff Pro 3.4.

Absolute Uninstaller

Размер: 1,63 Мбайт.

Статус: коммерческая.

Страница программы: http://www.glarysoft.com/absolute-uninstaller/.

Это программа, заменяющая собой стандартный диалог Windows Установка и удаление программ. Она использует для формирования списка установленных программ те же методы, что и диалог Установка и удаление программ (в начале книги об этих методах уже было сказано), но работает при формировании списка программ быстрее стандартного диалога.

Программа позволяет выполнять очистку диска от временных файлов, а также очистку реестра.

CCleaner (Crap Cleaner)

Размер: 1,37 Мбайт.

Статус: коммерческая.

Страница программы: http://www.ccleaner.com/.

Если для предыдущей программы очистка диска и реестра являются дополнительными возможностями, то у этой программы это основные функции. При этом данная программа, как считается, отлично с ними справляется. По традиции, она также представляет альтернативу стандартному диалогу Установка и удаление программ.

Хотелось бы подвести следующий итог рассмотрения этой программы и любых других, выполняющим очистку реестра или диска. Не стоит соглашаться со всем, что говорят эти программы, особенно если они предлагают удалить 100–200 «ненужных» параметров реестра. Как правило, результат такого удаления непредсказуем, но обычно совсем не такой, на какой рассчитывали вы и разработчики программы. Хотя, если вы будете использовать подобные программы при установленной программе SchadowUser Pro, речь о которой пойдет дальше, тогда о возможных последствиях можно не волноваться. Если, конечно, система сможет хотя бы загрузиться после использования чистильщиков системы.

PerfectDisk

Размер: 6,22 Мбайт.

Статус: коммерческая.

Страница программы: http://www.raxco.com/products/perfectdisk2k/.

Вы еще не забыли о стандартной оснастке Windows Дефрагментация дисков? Если нет, то предлагаю сравнить ее работу с работой этой программы. Приложение PerfectDisk также предназначена для фрагментации дисков. Работа с ней похожа на работу с оснасткой Windows — также возможно выполнить анализ фрагментации диска перед началом дефрагментации, также можно просмотреть отчет о состоянии диска. В общем, трудностей с ней быть не должно. Работает программа быстро — раздел размером в 10 Гбайт с уровнем фрагментации 16 % она дефрагментировала примерно за две минуты.

Effective File Search

Размер: 750 Кбайт.

Статус: коммерческая.

Страница программы: http://www.sowsoft.com/products.htm.

Еще одна альтернатива. На этот раз альтернатива стандартному диалогу поиска Windows. Программа может все, что и стандартный поиск Windows. Но, кроме этого, она умеет искать по шестнадцатеричному коду в файле (очень удобно, раньше это можно было сделать только с помощью Visual Studio .NET, загрузив в него все файлы, в которых нужно что-то искать), а также искать в найденном.

Easy Autorun Creator

Размер: 2,2 Мбайт.

Статус: коммерческая.

Страница программы: http://aw-software.com/products/eac.htm.

Комментарий: очень простая и удобная программа с минимальными возможностями, но красивым результатом.

Помните, когда в книге рассматривались возможности INF-файлов, был описан такой файл, как `autorun.inf`? С его помощью можно было настроить возможности автоматического запуска созданного вами диска после установки его в привод: указать значок для диска, файл, который будет выполняться после установки диска в привод и т. д. Именно подобный файл и можно создать с помощью программы Easy Autorun Creator 2.0. Но, кроме файла `autorun.inf`, она создает файл оболочки `autorun.exe`, который и запускается после установки диска в привод. После запуска `autorun.exe` перед вами отобразится диалог, содержащий список всех файлов, находящихся на данном диске (естественно, кроме файлов, создаваемых Easy Autorun Creator 2.0).

Однако как же работать с этой программой? После ее запуска перед вами появится диалог, в котором нужно указать имя проекта и путь к каталогу, в котором расположено содержимое, записываемое на диск. В дальнейшем имя проекта будет указано как в строке заголовка новой оболочки (хотя после создания `autorun.exe` строку заголовка можно изменить, просто изменив строку `caption` в созданном файле `autorun.inf`), так и в самой оболочке в виде текста. Оболочка `autorun.exe`, как правило, состоит из трех блоков: рисунка в верхнем левом углу, текста в правом верхнем углу и списка содержимого диска под рисунком и текстом.

После указания имени проекта и каталога, в котором он находится, щелкните кнопкой мыши на кнопке Autorun Settings, после чего перед вами появится новый диалог. В нем можно настроить значок диска, применяемый для создаваемой оболочки, заставку, размеры окна оболочки, а также музыкальный файл, воспроизводимый при запуске `autorun.exe`.

ПРИМЕЧАНИЕ

Не указывайте очень длинное имя проекта, так как оно будет плохо смотреться в самой оболочке.

Вот и все возможности этой программы. После их указания нажмите кнопку Build, и программа создаст файл оболочки для содержимого указанной папки. После создания оболочки появится кнопка Test Autorun, нажатие которой приведет к запуску только что созданной оболочки.

Подведем итоги. Очень простая программа с минимальными возможностями. Но, несмотря на это, получаемый результат довольно качественно и красиво оформлен. Хотя есть и недостатки. Например, если указать слишком длинный текст в качестве имени проекта, то он будет располагаться неправильно (максимум в двух строках), поэтому описание содержимого диска в Easy Autorun Creator 2.0 создать нельзя. С ее помощью также нельзя создать отдельное описание для каждой программы, содержащейся на диске. Хотя это можно выполнить уже после создания файла `autorun.exe`, если, конечно, вы раньше создавали HTML-страницы. Для этого достаточно отредактировать файл `eac.htm`, создаваемый в каталоге вместе

с файлом `autorun.exe` — именно `eac.htm` является стартовой страницей оболочки, открываемой файлом `autorun.exe`.

Inno Setup

Размер: 1,12 Мбайт.

Статус: бесплатная.

Страница программы: http://www.jrsoftware.org/.

Комментарий: отличная программа, да еще и бесплатная.

Программа предназначена для создания файлов-установщиков (ISS). Она проста в использовании и работе и создает стандартные программы установки (даже с возможностью выбора русского языка).

Essential NetTools

Размер: 3,2 Мбайт.

Статус: коммерческая.

Страница программы: http://www.tamos.ru/products/nettools/.

Комментарий: хорошая программа с интересными возможностями.

Программа предназначена для мониторинга сетевой активности и позволяет выполнить следующие действия:

- просмотреть список файлов, обращающихся к портам компьютера;
- просмотреть все запущенные в данный момент процессы и диаграмму загрузки процессора ими;
- выполнить пинг, сканирование (и прослушивание) портов или трассировку пути к удаленным компьютерам;
- просмотреть список общедоступных ресурсов компьютера.

NetLimiter

Размер: 2,3 Мбайт.

Статус: коммерческая.

Страница программы: http://www.netlimiter.com/.

Еще одна программа для работы с сетью. С помощью этой программы можно следить за количеством трафика, генерируемым каждым процессом системы. Но этим функции NetLimiter не ограничиваются — с ее помощью можно наложить квоты для отдельных процессов на закачиваемый объем данных.

Google Web Accelerator

Размер: 1,35 Мбайт.

Статус: бесплатная.

Страница программы: http://webaccelerator.google.com/.

Довольно интересная программа от Google. Как утверждают разработчики, она ускоряет открытие веб-страниц. Это действительно так, поэтому привожу адрес страницы, посвященной методу работы этой программы: http://webaccelerator.google.com/support.html.

Everest Ultimate Edition 2006

Размер: 5,55 Мбайт.

Статус: коммерческая.

Страница программы: http://www.lavalys.com/.

Хотите узнать больше о своем компьютере? Тогда эта программа для вас. С ее помощью вы узнаете все или почти все о своей материнской плате, процессоре, чипсете, BIOS, оперативной памяти и многом другом.

SpeedFan

Размер: 1,38 Мбайт.

Статус: коммерческая.

Страница программы: http://www.almico.com/speedfan.php.

Программа предназначена для мониторинга температуры процессора, жесткого диска и т. д. Хотя полный список ее функций чуть шире: определение частоты вращения вентиляторов, температуры системных компонентов, вольтажа системных компонентов, а также мониторинг загруженности процессора.

TaskInfo

Размер: 1,3 Мбайт.

Статус: коммерческая.

Страница программы: http://www.iarsn.com/taskinfo.html.

Еще одна программа для мониторинга состояния системы. Она является аналогом стандартного Диспетчера устройств Windows, но аналогом очень «продвинутым». Программа может следить за нагрузкой на следующие компоненты компьютера: процессор, оперативную память, виртуальную память, передаваемые и получаемые по сети данные, а также за скоростью считывания\записи данных на диск.

Еще одним плюсом программы является возможность просмотра информации о запущенных процессах — можно не только узнать, какой процесс запущен в данный момент, но и посмотреть, какие файлы использует тот или иной процесс, работает ли он с сетью, какие функции, библиотеки использует и т. д. И это далеко не все ее возможности.

Еще одним плюсом программы является значок в области уведомлений. Если значок Диспетчера задач отображает загруженность только процессора, то значок программы TaskInfo показывает загруженность каждого компонента системы.

Security Task Manager

Размер: 1,45 Мбайт.

Статус: коммерческая.

Страница программы: http://www.neuber.com/taskmanager/index.html.

Это альтернатива Диспетчеру задач. С помощью этой программы можно просмотреть список запущенных процессов. При этом ее особенностью является рейтинг опасности запущенных процессов. Другими словами, программа проверяет, следит ли процесс за действиями пользователя, перехватывает ли нажатия клавиш или ответы браузера и т. д., и на основе полученных данных строит список запущенных процессов — чем выше процесс в этом списке, тем он опасней с точки зрения программы. Она также отображает строки текста, встречающиеся в коде файла процесса, и может изолировать опасный процесс.

Safe'n'Sec

Размер: 12,7 Мбайт.

Статус: коммерческая.

Страница программы: http://www.star-force.com/.

Если вы серьезно относитесь к безопасности компьютера, то вам может понадобиться программа, автоматически следящая за процессами и сообщающая вам об их подозрительных действиях. Такой программой и является Safe'n'Sec. Она сообщает вам о различных попытках процессов запустить или завершить какой-либо дочерний процесс, а также следит за сетевой активностью (на предмет хакерских атак). Программа также позволяет выполнить поиск вредоносных программ на жестком диске компьютера.

Punto Switcher

Размер: 304 Кбайт.

Статус: бесплатная.

Страница программы: http://punto.ru/switcher/.

Маленькая, но просто незаменимая программа, если вы часто пользуетесь текстовым редактором. Она автоматически распознает вводимый пользователем текст и определяет, правильная ли раскладка установлена в момент ввода. Если вы пытаетесь, например, написать слово «например» при помощи английской раскладки, то программа автоматически преобразует английскую раскладку в русскую.

Программа также определяет ошибки ввода (не синтаксические). Например, если вы введете слово «ПРивет» в качестве названия папки в Проводнике Windows (да, программа следит даже за вводом текста в оболочке Windows), то она автоматически изменит название папки на «Привет».

TaskSwitchXP Pro

Размер: 357 Кбайт.

Статус: коммерческая.

Страница программы: http://www.ntwind.com/.

Еще одна маленькая программа, которая сможет облегчить вам жизнь. Она является аналогом стандартного диалога Windows, вызываемого комбинацией клавиш Alt+Tab. Особенностью программы является ее снимок, отображаемый напротив названия при вызове диалога с помощью Alt+Tab. Это может быть полезно, когда вы работаете сразу с несколькими файлами одного приложения, например с десятью окнами Internet Explorer.

TrayIt!

Размер: 150 Кбайт.

Статус: бесплатная.

Страница программы: http://www.teamcti.com/trayit/trayit.htm.

С помощью этой маленькой программы можно поместить в область уведомлений многие программы, которые по умолчанию помещаться туда совсем не хотят, занимая лишнее место на Панели задач. Работать с этой программой очень просто. После ее установки перед вами отобразится диалог, в котором можно выполнить настройку TrayIt!, хотя это и необязательно. Если вы настраивать программу не собираетесь, то закрывайте диалоговое окно (кнопка Continue Using Tray IT). Теперь самое интересное. Если вы используете Проигрыватель Windows Media, то можете поэкспериментировать с ним, иначе выбирайте любую другую программу. Щелкните правой кнопкой мыши на кнопке сворачивания программы (в строке заголовка), после чего перед вами появится контекстное меню программы TrayIt!. В этом меню выберите команду Play is System Tray, после чего щелкните кнопкой мыши на кнопке сворачивания программы. Если вам повезет, то программа свернется в область уведомлений и теперь после щелчка на кнопке будет делать это всегда.

QDictionary

Размер: 1,96 Мбайт.

Статус: бесплатная.

Страница программы: http://www.anplex.ru/.

Программа является англоязычным словарем. Еще одним, воскликнете вы, нам и так хватает Lingvo. Но все не совсем так — особенностью этой программы является моментальный перевод слова после двойного щелчка на нем. Иными словами, вам не придется выделять слово, запускать словарь, помещать в поле для перевода слово из буфера обмена и т. д.

SlyControl

Размер: 4,35 Мбайт.

Статус: коммерческая. Однако для русскоговорящих пользователей разработчик решил сделать программу бесплатной — для регистрации на месяц нужно просто ввести на русском языке название текущего месяца. Для следующей регистрации на месяц нужно ввести название следующего месяца.

Страница программы: http://slydiman.narod.ru/.

Если вы имеете пульт дистанционного управления (ДУ), то эта программа вам может пригодиться. Благодаря ей вы сможете управлять работой компьютера с помощью пульта: это включение\выключение компьютера, работа с аудио и видео и многое другое. Рекомендуется посетить сайт разработчика, где вы сможете скачать дополнительные сценарии для работы пульта с компьютером.

Tag&Rename

Размер: 2,56 Мбайт.

Статус: коммерческая.

Страница программы: http://www.softpointer.com/tr.htm.

Программа предназначена для редактирования тегов музыкального файла. С ее помощью можно как переименовывать файлы на основе их тегов, так и редактировать сами теги. Конечно, сейчас даже лицензионные музыкальные диски не содержат никаких тегов, поэтому особенно актуальной является возможность работы данной программы сразу с несколькими файлами, то есть выделяете сразу все файлы одного альбома, а потом редактируете тег, например название альбома.

MultiSet

Размер: 1,95 Мбайт.

Статус: коммерческая.

Страница программы: http://www.almeza.com/.

Интересная программа, с помощью которой можно автоматизировать процесс установки программ. Работает она следующим образом: если вам необходимо установить какую-либо программу на компьютер, то вы запускаете MultiSet и в меню Файл выбираете команду Новый проект. После этого начнется процесс установки новой программы с вашим участием — вы устанавливаете программу, а MultiSet следит за вашими действиями и записывает их в свою базу данных. В будущем, если вам еще раз придется устанавливать эту программу, вы сможете просто выделить ее в списке MultiSet и запустить ее автоматическую установку. Больше делать вам ничего не придется — MultiSet выполнит установку точно так же, как раньше ее выполняли вы.

NikSaver

Размер: 2,01 Мбайт.

Статус: коммерческая.

Страница программы: http://www.niksaver.com/rus/.

Если предыдущая программа позволяла автоматизировать процесс установки программ, то программа NikSaver позволяет сохранить настройки различных программ в REG-файлах, после чего их можно будет импортировать на другой компьютер или в операционную систему для быстрого восстановления настроек программы. На данный момент NikSaver поддерживает не слишком большой список программ, но успешно развивается.

DocRepair

Размер: 726 Кбайт.

Статус: коммерческая.

Страница программы: http://www.jufsoft.com/.

Программа для восстановления поврежденных файлов Microsoft Word. Если после какого-либо события, например после отключения света, DOC-файл, с которым вы работали, не хочет запускаться, а Microsoft Word пишет о том, что файл поврежден, то воспользуйтесь программой DocRepair. Скорее всего, она сможет восстановить поврежденный файл.

Password Door

Размер: 791 Кбайт.

Статус: коммерческая.

Страница программы: http://www.toplang.com/.

Если необходимо защитить запуск какой-нибудь программы паролем, то эта программа для вас. После установки пароля на программу пользователь, не знающий его, не только не сможет запустить ее, но также не сможет ее удалить или переместить.

ShadowUser

Размер: 6,29 Мбайт.

Статус: коммерческая.

Страница программы: http://www.shadowstor.com/.

Вам когда-нибудь хотелось поиздеваться над Windows? Например, после появившегося «синего экрана» удалить ненужные ей (конечно, Windows так не считает) файлы или параметры реестра? Если такое желание возникало, но вас сдерживала природная лень (не хотелось потом переустанавливать операционную систему), то теперь вас ничто не сдерживает. После установки этой программы вы можете делать с Windows все, что угодно, например подкинуть ей несколько вирусов или удалить «ненужные» параметры реестра. А когда вы насладитесь издевательством над Windows, вам ничто не помешает вернуть систему к предыдущему состоянию с помощью программы ShadowUser (главное, не переусердствовать с издевательствами, чтобы Windows смогла загрузиться). Программа ShadowUser именно для этого и предназначена.

Работать с программой просто. После ее установки необходимо перезагрузиться, после чего в меню Mode программы выбрать команду Activate ShadowMode. Далее нужно еще раз перезагрузиться, чтобы был задействован соответствующий режим (в окне программы напротив строки ShadowMode появится надпись Enabled). Теперь можете делать с системой или своими файлами все, что хотите. Если после некоторых экспериментов вы захотите восстановить предыдущее состояние системы, то просто с помощью меню Mode отключите теневой режим, выбрав перед этим необходимое действие: либо указав программе полностью удалить все изменения, которые произошли в системе во время активности режима ShadowMode, либо приказав программе сохранить все сделанные изменения.

База параметров реестра

Как уже говорилось в книге, на диске также содержится база данных Access, содержащая сведения более чем о 2500 параметрах и ветвях реестра.

Обратите внимание, что данная база является авторской разработкой и ее распространение, продажа или использование в своих проектах преследуется законом об авторском праве.

ПРИМЕЧАНИЕ

На данный момент база данных не является готовым продуктом: она будет дорабатываться. Если вы найдете ошибки в описаниях параметров, желаете добавить какие-либо параметры или хотите поучаствовать в разработке интерфейса базы, пишите по электронному адресу parazone@mail.ru.

После открытия базы данных появится окно с информацией о новых функциях, которые будут реализованы в следующих версиях базы данных, а также изображения экрана некоторых готовых стилей оформления базы (если у вас есть идеи по оформлению будущих версий базы данных, присылайте их по адресу parazone@mail.ru). Чтобы перейти к окну поиска, необходимо дважды нажать кнопку Ok: после первого нажатия форма очистится, а после второго — откроется форма поиска.

Форма поиска содержит фильтры, позволяющие осуществить поиск в базе данных. Можно воспользоваться фильтром по версии Windows, в которой может использоваться данный параметр (флажки Фильтр Windows XP и Фильтр Windows 2000), по типу параметра (флажок Фильтр по типу) или по одному из ключей поиска (флажки Фильтр по ключу и Фильтр по программе). Можно также выполнить поиск по части названия ветви реестра, в которой находится параметр, или по части названия параметра (флажок Фильтр по части ветви или параметра), или по описанию параметра (флажок Фильтр по описанию записи). Форма поиска поддерживает множественные фильтры, то есть, например, одновременный поиск по версии Windows, типу параметра и части названия параметра. Пока нельзя произвести поиск по нескольким элементам одного списка, например одновременный поиск параметров DWORD-типа и строкового, но в следующей версии базы он будет добавлен. Нельзя также выполнить поиск, исключающий какие-то условия (например, поиск всех записей, не имеющих DWORD-тип).

После формирования запроса (установки соответствующих флажков и выбора в списках или полях нужного элемента) необходимо нажать кнопку Поиск. Начнется поиск записей в базе и вывод результата (для этого используется отдельная форма).

ВНИМАНИЕ

Информация, содержащаяся в базе, может устареть или оказаться неправильной, поэтому следует несколько раз подумать перед тем, как довериться тому, что написано в базе. По этой причине советую обновлять версию базы.

Форма вывода результата включает в себя названия ветви реестра и параметра, удовлетворяющего поиску, а также описание этого параметра и его значение в вашем реестре (два поля в нижней части экрана). С помощью данной формы можно установить новое значение параметра (кнопка Edit напротив одного из значений параметра в вашем реестре) или удалить существующий параметр (кнопка Del напротив одного из значений параметра в вашем реестре). Для перехода между найденными записями (на экран выводятся сведения только об одном параметре) используется стандартная панель перехода Access (в нижней части формы), содержащая кнопки перехода, а также отображающая количество найденных записей и номер текущей записи.

ПРИМЕЧАНИЕ

В данной версии базы нельзя отредактировать значение параметра или узнать его значение в вашей системе, если ветвь, в которой находится параметр, является общей (то есть хранит подразделы, имеющие общие названия, например «GUID-номер учетной записи»).

Чтобы опять перейти к форме ввода запроса поиска, нужно нажать кнопку Поиск в верхнем левом углу. Для выхода из базы следует нажать кнопку Выход.

Листинги

В папке Листинги приведены листинги, рассмотренные в книге.